The use of behavioural kinematics to decode affective dynamics in naturalistic paradigms

by

Austin Varughese

A dissertation submitted in partial fulfillment

of the requirements for the degree of

MASTER'S OF SCIENCE IN NEUROSCIENCES

at the

NATIONAL BRAIN RESEARCH CENTRE

June 2025

ब्रिक – राष्ट्रीय मस्तिष्क अनुसंधान केंद्र
BRIC – National Brain Research Centre
(जैव प्रौद्योगिकी अनुसंधान एव नवाचार परिषद, भारत सरकार)
एन० एच०–08, मानेसर–122 052 (हरियाणा), भारत
(An Institute of the Biotechnology Research and Innovation Council, Govt. of India)
NH-08, Manesar - 122 052 (Haryana), INDIA

CERTIFICATE

This is to certify that the dissertation report entitled '**The use of behavioural kinematics to decode affective dynamics in naturalistic paradigms**' is the result of work carried out by **Austin Varughese** in the Division of **Cognitive Brain Dynamics Laboratory**, National Brain Research Centre, Manesar, Haryana, India, for the fulfilment of the requirements for the M.Sc. in Neuroscience's degree.

The work presented herein is original and has not been submitted previously for the award of any degree or diploma to **National Brain Research Centre (Deemed to be University)** or to any other University. This work is based on the guidelines of **National Brain Research Centre (Deemed to be University)** and is a record of the candidate's own efforts.

Dr. Arpan Banerjee
(Supervisor)

(Director, NBRC)

Place: Manesar
Date: 2/6/2025

दूरभाष / Telephone : +91-124-2845200 फैक्स / Fax : +91-124-2338910
ई-मेल / E-mail : info.nbrc@nbrc.ac.in Website : www.nbrc.ac.in

DrillBit

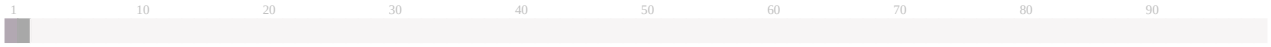
The Report is Generated by DrillBit Plagiarism Detection Software

Submission Information

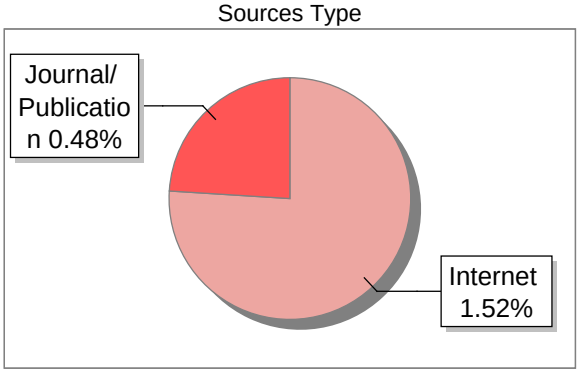
Author Name	Austin Varughese
Title	The use of behavioural kinematics to decode affective dynamics in naturalistic paradigms
Paper/Submission ID	3930196
Submitted by	ddlal@nbrc.ac.in
Submission Date	2025-06-23 16:27:57
Total Pages, Total Words	51, 10936
Document type	Thesis

Result Information

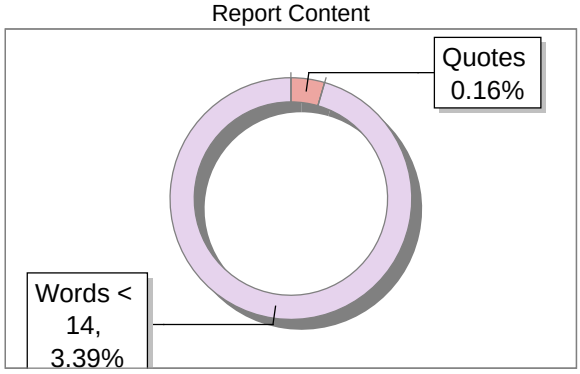
Similarity **2 %**



Sources Type



Report Content



Exclude Information

Quotes	Excluded
References/Bibliography	Excluded
Source: Excluded < 14 Words	Excluded
Excluded Source	**0 %**
Excluded Phrases	Excluded

Database Selection

Language	English
Student Papers	Yes
Journals & publishers	Yes
Internet or Web	Yes
Institution Repository	Yes

A Unique QR Code use to View/Download/Share Pdf File

DECLARATION BY THE CANDIDATE

I, Austin Varughese, hereby declare that the work presented in this dissertation is carried out by me, under the guidance of Dr. Arpan Banerjee at the National Brain Research Centre, (Deemed University), Manesar, Haryana, India.

I also declare that no part of this dissertation has been previously submitted for the award of any degree or diploma at National Brain Research Centre (Deemed University) or any other University.

Austin Varughese

The use of behavioural kinematics to decode affective dynamics in naturalistic paradigms

by

Austin Varughese

Submitted to the Cognitive Brain Dynamics Laboratory
on June 15, 2025 in partial fulfillment of the requirements for the degree of

MASTER'S OF SCIENCE IN NEUROSCIENCES

ABSTRACT

Emotions are an important part of our socio-cognitive toolkit, they help us prioritise our attention resources and guide our decision-making processes, resulting in more apt responses. A great deal of success in social contexts could be attributed to our ability to use our emotions to better fit our responses to the environment. Emotion regulation is an important aspect of affective behaviours, and has an important role in the investigation of psychopathological states. It is our inherent ability to modulate our emotional responses to better fit the needs of the environment. Affective flexibility is the ability to transition across various emotional states while engaging in a task. Consequently, having a mechanistic understanding of these phenomena is crucial to understanding emotional dysregulation as seen in various neuropsychiatric disorders. A major challenge in studying these phenomena has been the absence of a robust measure of the emotional flux of an individual. Traditional self-report questionnaires are susceptible to being biased by the subject's own percept of their mental state, and lack the granularity to capture quite many features necessary to study these phenomena. We propose a novel measure derived from the experience sampling method and actigraphy, allowing for the capture of the emotional flux of an individual through their motion on a two-dimensional emotional space. Furthermore, we demonstrate that this measure can be used to study affective flexibility, effectively capturing individual variability in emotional regulation.

Thesis supervisor: Prof. Dr. Arpan Banerjee
Title: Scientist VII

Acknowledgments

I would like to begin by expressing my gratitude to my advisor Prof Dr Arpan Banerjee for his invaluable guidance and support. Discussing my ideas with him and receiving critical feedback was an invaluable tool in troubleshooting the countless challenges I have faced during this project. Thank you for you patience and support throughout this journey.

I would also like to thank Shashwati Tripathi, a collaborator and support system. Her invaluable insights, unwavering faith in my hyperactivity, and tolerance of my ability to incessantly talk about my ideas were crucial to the success of this project.

I am grateful to my colleagues, Dhrubajoti, Vinitha, Azman, and Tanmay for their valuable inputs and continued support. I am grateful to all members of the lab for ensuring that I was never alone in my journey at the lab.

My heartfelt gratitude to my friends Vibha, Atharva and Lisha for continuous encouragement and support. Through countless midnight discussions, obscenely bad jokes, and the occasional existential crisis they have helped add colour to an otherwise monochrome journey.

Finally, I would to thank my mother and sister, their struggles and sacrifices have been the foundation of my journey. Without their support, I would not be where I am today. Their faith in my ambitions and dreams has been the fire that has kept me going through even the most unkind times. I am forever grateful for their presence in my life.

Contents

List of Figures 9

1 Introduction **10**

1.0.1 Models of emotion regulation . . . 11
1.0.2 Affective flexibility . . . 13
1.0.3 Continuous capture of emotional flux . . . 13
1.0.4 Goals of this study . . . 14

2 Materials and Methods **15**

2.1 Participant details . . . 15
2.2 Experiment paradigm . . . 16
2.2.1 Behavioural data . . . 16
2.2.2 Anatomical data preprocessing . . . 17
2.2.3 Functional data preprocessing . . . 17
2.2.4 Susceptibility correction . . . 18
2.2.5 Denoising neural data . . . 19
2.2.6 Correlation analysis . . . 19
2.2.7 General Linear Models . . . 20

3 Results **21**

3.1 Using kinematics to glimpse into the emotional brain . . . 22
3.2 Deriving emotional reactivity from kinematics . . . 24
3.2.1 Calculating emotional reactivity . . . 24
3.3 Kinematics reflect trait induced individual variability . . . 26
3.3.1 Median emotional reactivity is influenced by trait features . . . 27
3.3.2 Presence of neuropsychiatric traits negatively impacts emotional reactivity . . . 28
3.4 Neural data validates affective features captured by kinematics . . . 30

4 Discussion **32**

A Code listing **35**

References 62

List of Figures

1.1 Process Model of Emotion Regulation; Recreated from James J Gross 1998b . . . 11
1.2 Extended process Model of Emotion Regulation; Recreated from James J. Gross 2015 . . . 12

3.1 Group level responses strongly resemble (rho ≅ 0.59, p ≅ 0) the valence values proposed in Mathur et al. 2015, validating stimulus strength and paradigm design, Recreated from: Tripathi et al. 2025. . . . 21
3.2 Raster plot of emotional transitions in an individual derived by binarisation of the acceleration time series . . . 23
3.3 Correlation between emotional spread and trait scores . . . 24
3.4 Correlation between number of shifts and trait scores . . . 24
3.5 Deriving emotional response time from kinematic features . . . 25
3.6 Response latencies across various ragas . . . 26
3.7 Distribution of trait scores for all participants used within this study . 26
3.8 Emotionality potentially impacts emotional reactivity . . . 27
3.9 Median emotional reactivity is not affected by subclinical depressive and anxious behaviours . . . 27
3.10 Subclinical neuropsychiatric behavioural traits don't influence emotional reactivity in positive to negative valence transitions . . . 28
3.11 Depressive and anxious behaviours significantly affect emotional reactivity when transitioning away from negative valence states . . . 29
3.12 Trait features have no influence in + to - valence transitions . . . 29
3.13 Trait features have no influence in - to + valence transitions . . . 30
3.14 Z scored stat maps obtained from the 2nd level GLM . . . 31

Chapter 1

Introduction

Emotions are precisely defined internal states induced by internal or environmental cues (Anderson and Adolphs 2014). They are an important part of our socio-cognitive toolkit, and as social animals, they are imperative to our success within society.

Traditional approaches to studying emotions have depended upon and utilised visual cues, especially faces. This capitalises on our ability to immediately capture changes in the emotional state of an individual through their facial features (Globisch et al. 1999; Vuilleumier, Armony, et al. 2001; Vuilleumier, Richardson, et al. 2004). The strong response that emotional faces generate, paired with the static nature of these paradigms helped us explore and investigate affective behaviours to a great extent. Yet, they fail to capture an important aspect of emotions, their dynamicity.

Our affective state is in constant flux, with new states being generated or current states being modulated in some fashion in response to the changes in our environment. The field of emotion dynamics attempts to study this flux, specifically how it fluctuates over time, the underlying mechanism, and it's consequences on other cognitive processes (Sander et al. 2005; Kuppens, Stouten, et al. 2009; Kuppens and Verduyn 2015).

This emotional flux allows for more effective utilisation of our limited attentional resources. Stimuli with higher affective value are immediately prioritised and responded to, allowing for near instantaneous prioritisation of salient features in our environment (Bradley et al. 1997; Vuilleumier and Schwartz 2001). This dynamicity is an important tool in social contexts where our priorities need to be updated in real-time. Dependent on the temporal scale at which we study this flux, we can classify these behaviours into subgroups. Emotions work at smaller timescales as an acute and defined response, while moods are more defuse in nature and lack clear distinction or definitions (James J Gross 1998b). These states constantly influence our cognition and modulate our responses to better fit to the situation requirements.

1.0.1 Models of emotion regulation

Responding emotionally to every change in our environment may not be in our interests too. A great example of this would be the modern-day workplace, the actions of someone in a position of authority might make us angry, yet we control and alter our emotions to protect ourselves. This ability to change the emotional narrative of the environment to better suit our needs and cope to the stressors is called ***emotion regulation*** (James J Gross 1998a; James J Gross 1998b).

The process model of emotion regulation proposed by James J Gross 1998b was the first model to provide a mechanistic insight into the processes that underlie emotion regulation, and has been extensively used for the study of emotion regulation. This model proposed a simple explanation of how perceived cues (*internal* or *external*), lead to the generation of an emotional response when attended to, resulting in a response being generated that is modulated by our emotional state to best fit the context.

In ***Fig:*** 1.1, we can see these steps as the primary chain of events. In response to new information (*situation*) being perceived (*attention*), we evaluate (*appraise*) it to generate an affective response (*response*). These response can further act as modulators influencing new percepts. At each step of this process through various regulatory strategies ***Fig:*** 1.1 we can alter our emotional response.

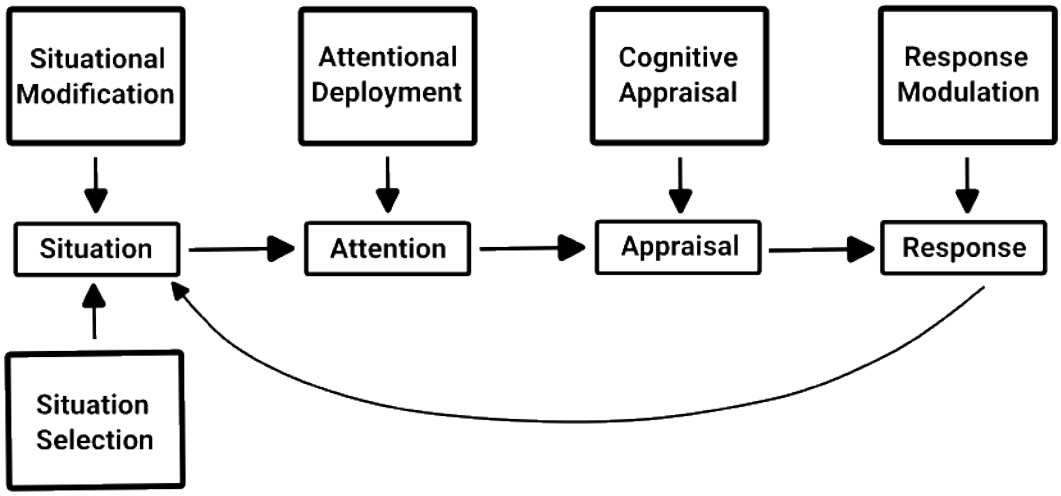


Figure 1.1: Process Model of Emotion Regulation; Recreated from James J Gross 1998b

This also brings to light the immense potential of emotions in modulating our cognitive processes. infinitesimal deviations or deficits in any of these processes can lead to the manifestation of a spectra of pathopsychological behaviours.

While the process model was quite exhaustive, it lacked the ability to elaborate on the link between emotion regulation and generation. The extended process model of emotion regulation filled this gap, answering a long standing question. It provided a

generic framework that allowed for the unification of emotion regulation with generation, elaborating them to be hierarchical appraisal systems. With a more mechanistic outlook on the processes defined in the original model, it provided an avenue to better explore variability in affective behaviours, and emotional dysregulation (James J. Gross 2015). Furthermore, it drew a clearer relationship between emotion regulation and time.

Within the extended model, there are 4 broad stages to the generation of a response, first is the emergence of a need for a response (*World/W*), this is followed by the perception of this requirement by allotment of our attentional resources (*Perception/P*) and an evaluation (*Valuation/V*) of the necessity and cost of a response (*Action/A*). This leads to the initiation of a new cycle which determines the best avenue to respond to the situation (*World/W*) with, eventually ending the decision-making process by the initiation of a cognitive response. Through time, each of these stages constantly updates the affective state of the individual, culminating in the continuous flux of emotions one experiences (James J. Gross 2015).

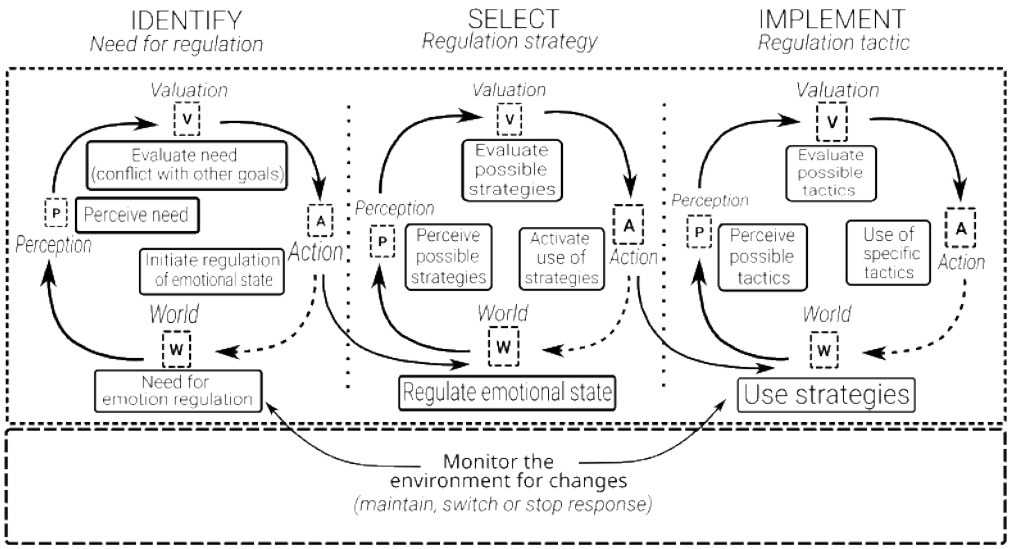


Figure 1.2: Extended process Model of Emotion Regulation; Recreated from James J. Gross 2015

This put emphasis on the continuous and context-sensitive nature of our emotions, eclipsing the traditional outlook of studying them as static events arrested within windows of time. Furthermore, it provided an avenue to connect emotional regulation with other cognitive processes, specifically, executive functions. The most significant among them being our ability to adapt our thoughts and behaviours in response to the ever changing environment, ***behavioural and cognitive flexibility***.

1.0.2 Affective flexibility

Cognitive flexibility is our ability to transition between engaging in multiple streams of thought while engaging in a situation, as per the needs of the situation. Behavioural flexibility, while similar to cognitive flexibility, is the behavioural adaptation to the ever changing needs of the environment. Flexibility as a whole falls under a broader family of cognitive phenomena necessary for goal-oriented actions called executive functions (Uddin 2021).

In a fashion similar to the component theory, studies have broadly classified 3 components constituting executive functions, ***mental set-shifting***, ***information updation in the working memory***, and ***inhibition of prepotent actions***. A closer observation would point us towards the stark similarities between executive functioning and emotional regulation. Mental set-shifting is our ability to move across various chains of thoughts, ergo, cognitive flexibility.

While cognitive flexibility has been extensively studied through the lens of an attentional process, utilising the Wisconsin Card Sorting task and Trail marking tests, there has been limited exploration of its relationship with affect and emotional dynamics. This was interesting because paired with the extended process model, one could study affective phenomena through the quantification of flexibility in behaviour. This lead to Genet et al. 2013 coining the term ***affective flexibility***, the ability to transition across various affective states in accordance with the needs of the environment.

1.0.3 Continuous capture of emotional flux

To study emotions dynamically we would need to establish a framework that allows for monitoring of the emotional state in real-time, while also allowing us to study the underlying mechanisms that support these behaviours. This implies that we would require a task that allows for continuous capture of the emotional state of an individual. While affective studies have never approached emotions through a continuous capture technique, the study of moods has a close analogue, the experience sampling method.

The experience sampling method (ESM) (Csikszentmihalyi et al. 1977) and ecological momentary assessment (EMA) (Bolger et al. 2003; Hoppmann and Ho 2015) are continuous capture techniques employed in the study of moods. These techniques involve the participant being required to fill out questionnaires that captured their affective state at that timepoint at fixed intervals, usually multiple times a day. These studies spanned weeks or even months, and provided a dataset that accurately quantified mood variability over time.

While these techniques were capable of capturing temporal variability in affect, they lacked the precision required to study acute changes underlying emotion regulation. Although, they did establish a framework that could be adapted to study finer changes

in affective states.

Tripathi et al. 2025 developed upon these techniques to create a continuous capture paradigm that used a naturalistic stimulus to capture the temporal variation of emotions in an individual. Pairing it with fMRI BOLD collection to capture the neural correlates of these behaviours. Thus, a continuous measure of emotional flux was established.

1.0.4 Goals of this study

The broad goal of this study was to study the temporal evolution of emotions at an individual level, and explore the source of individual variability in emotional regulation and affective flexibility. We aimed to employ the continuous capture dataset to explore various features of the participant behaviour that can be developed into a measure of affective flexibility.

The primary aim of this study was to develop a measure of affective flexibility that could be utilised in modelling emotional flux in response to a near - naturalistic environment.

Ambulatory assessment techniques(AA) are a subset of EMA that approaches the study of affect in a holistic manner, by capturing various features of participant behaviour and studying their relationship with the affective state of the individual. Actigraphy is one such technique that captures the kinematics of an individual for prolonged duration to study their behaviour. This has led to multiple studies observing and commenting upon lethargic phenotypes in children with depression (Aronen et al. 2011; Razavi et al. 2011). Although, such an approach has not been employed in the study of affective flexibility.

We hypothesised that employing a similar approach to this study would enable us to observe fine-grained changes otherwise imperceptible. Using the kinematics of the participant, we aimed to capture and characterise the transition between various affective states, a definite measure of individual emotional flux. We further hypothesised that these features would be heavily influenced by the personality traits of the individual. Thus, allowing us to peek at how the underlying mechanisms of affective flexiblity are modulated by predisposing factors.

Chapter 2

Materials and Methods

The emotional music dataset (Tripathi et al. 2025) is a task-based fMRI study that involves continuous rating of a naturalistic auditory stimuli on a 2D scale derived from the Russell's circumplex (Russell 1980). The stimulus was constructed by concatenating Hindustani ragas of varying affective values. The imperceptible boundaries of the ragas and their varying emotional content provided us with a range of emotional perturbations. We chose to pursue our study using this dataset due to the vividness and inherent variability in perception of emotional content of an auditory stimulus (Mathur et al. 2015). The absence of distinct visual cues in the study design compels the participant to evaluate their affective state attentively, and consequently it strongly amplifies trait induced variability in behaviour. Furthermore, the repertoire of questionnaires employed provided us with the information required to accurately characterise individual trait features. Finally, simultaneous capture of neural data enabled us to explore the neural correlates of behaviours under study.

2.1 Participant details

Individuals from the National Capital Region (NCR), India, ranging in age from 18 to 35 years of age were recruited. The study had a total of 53 participants, of which approximately 62.3% reported their gender as female and 37.8% reported their gender as male. All participants were right handed. All procedures were approved by the NBRC Institutional Human Ethics Committee (IHEC), and informed written consent from all subjects was collected prior to the experiment. The study was approximately 1.5 hours long, and the subjects received monetary compensation upon successful completion of the experiment.

For the behavioural data, 2 subjects were removed for reporting issues with stimulus presentation, and 5 subjects were removed for reporting prior diagnosis or history of

neuropsychiatric disorders. All behavioural analysis were performed on the remaining 46 participants, within whom 63.04% were females and 36.9% were males.

For the neural data, alongside the 7 subjects removed on the basis of behavioural markers, 12 more participants were removed for having excessive head motion parameters and not passing quality checks. All neural analysis were performed on the remaining 34 participants, within whom 61.8% were females and 38.2% were males.

2.2 Experiment paradigm

Neuroimaging data was acquired on a 3T Siemens® Prisma™ scanner with a 64-channel head-coil using a *T2** weighted echo planar imaging (**EPI**) pulse sequence (TR 1200ms, TE 39ms, flip angle of 50°, whole brain with 60 slices at 2mm slice thickness, inplane resolution of $2mm^2$ and FOV of $192mm^2$). Anatomical scans were acquired using a **T1-weighted FLAIR** sequence (TR 2200ms, TE 2.45ms, flip angle 8°, whole brain with 192 slices at 1mm thickness, in-plane resolution of $1mm^2$ and FOV of $244mm^2$). A fieldmap was acquired to correct for field inhomogeneities using a gradient echo fieldmap (**GRE-fieldmap**) sequence (TR 590ms, TE1 4.92ms, TE2 7.38ms, flip angle 78°, whole brain with 60 slices at 2mm slice thickness, in-plane resolution of $2mm^2$ and FOV of $192mm^2$). Audio was delivered using in-ear headphones, and a mirror fixed to the head coil was used to allow the participant to see the response screen. Subject responses were captured using a Current Designs® Tethyx™, an MR-safe joystick. Preprocessing was performed using fMRIprep 25.0.0 (Esteban et al. 2019; RRID:SCR-016216), which is a streamlined preprocessing pipeline based upon Nipype 1.9.2 (Gorgolewski et al. 2011; RRID:SCR-002502).

2.2.1 Behavioural data processing

The experimental paradigm was hosted using Presentation ™, which stored the responses as a log file for each subject. The raw data was sampled at approximately 60 Hertz, but was downsampled to 13.3 Hertz to overcome jitters in the data. For neural analyses this data was further downsampled to TR time (0.84 Hertz). We then rescaled the data from the original +/- 450 scale to a +/-1 scale. A dead-zone of 5% around the neutral position was applied to account for any jitters in the data due to inaccuracies in the response device.

2.2.2 Anatomical data preprocessing

T1-weighted (T1w) image found within the input BIDS dataset was corrected with **N4BiasFieldCorrection** (Tustison et al. 2010) for intensity non-uniformity (INU), distributed with ANTs 2.5.4 (Avants et al. 2008, RRID:SCR-004757), and used as T1w-reference throughout the workflow. The T1w-reference was then skull-stripped with a Nipype implementation of the **antsBrainExtraction** workflow (from ANTs), using OASIS30ANTs as target template. Brain tissue segmentation of cerebrospinal fluid (CSF), white-matter (WM) and grey matter (GM) was performed on the brain-extracted T1w using **fast** (FSL, RRID:SCR-002823, Zhang et al. 2001). Volume-based spatial normalisation to one standard space (**MNI152NLin2009cAsym**) was performed through nonlinear registration with **antsRegistration** (ANTs 2.5.4), using brain-extracted versions of both T1w reference and the T1w template. The following template was selected for spatial normalisation and accessed with **TemplateFlow** (24.2.2, Ciric et al. 2022): ICBM 152 Nonlinear Asymmetrical template version 2009c (Fonov et al. 2009, RRID:SCR-008796; TemplateFlow ID: MNI152NLin2009cAsym).

2.2.3 Functional data preprocessing

First, a reference volume was generated using a custom methodology of fMRIPrep for use in head motion correction. Head-motion parameters with respect to the BOLD reference (transformation matrices and six corresponding rotation and translation parameters) are estimated before any spatiotemporal filtering using **mcflirt** (FSL, Jenkinson, Bannister, et al. 2002). The BOLD reference was then co-registered to the T1w reference using **mri-coreg** (FreeSurfer), followed by **flirt** (FSL, Jenkinson and Smith 2001) with the **boundary-based registration** (Greve and Fischl 2009) cost-function. Co-registration was configured with six degrees of freedom. Several confounding time series were calculated based on the preprocessed BOLD: framewise displacement (FD), DVARS and three region-wise global signals. FD was computed using two formulations following Power (absolute sum of relative motions, Power et al. 2014) and Jenkinson (relative root mean square displacement between affines, Jenkinson, Bannister, et al. 2002). FD and DVARS are calculated for each functional run, both using their implementations in Nipype (following the definitions by Power et al. 2014). The three global signals are extracted within the CSF, the WM, and the whole-brain masks. A set of physiological regressors was extracted for component-based noise correction (CompCor, Behzadi et al. 2007). Principal components are estimated after high-pass filtering the preprocessed BOLD time-series (using a discrete cosine filter with 128s cut-off) for the two CompCor variants: temporal (tCompCor) and anatomical (aCompCor). The tCompCor components were calculated from the top 2% variable voxels within the brain mask. For aCompCor, three probabilistic masks (CSF, WM and combined CSF+WM) are

generated in anatomical space. The implementation differs from that of Behzadi et al. in that instead of eroding the masks by 2 pixels on BOLD space, a mask of pixels that likely contain a volume fraction of GM is subtracted from the aCompCor masks. This mask is obtained by thresholding the corresponding partial volume map at 0.05, and it ensures components are not extracted from voxels containing a minimal fraction of GM. Finally, these masks are resampled into BOLD space and binarised by thresholding at 0.99 (as in the original implementation). Components are also calculated separately within the WM and CSF masks. For each CompCor decomposition, the k components with the largest singular values are retained, such that the retained components time series are sufficient to explain 50% of the variance across the nuisance mask (CSF, WM, combined, or temporal). The remaining components are dropped from consideration. The head-motion estimates calculated in the correction step were also placed within the corresponding confounds file. The confound time series derived from head motion estimates and global signals were expanded with the inclusion of temporal derivatives and quadratic terms for each (Satterthwaite et al. 2013). Additional nuisance timeseries are calculated using principal components analysis of the signal found within a thin band (crown) of voxels around the edge of the brain, as proposed by (Patriat et al. 2017). All resamplings can be performed with a single interpolation step by composing all the pertinent transformations (i.e. head-motion transform matrices, susceptibility distortion correction when available, and co-registrations to anatomical and output spaces). Gridded (volumetric) resamplings were performed using **nitransforms**, configured with cubic B-spline interpolation.

2.2.4 Susceptibility correction

The spatial localisation of the MR signal is regulated by the magnetic field gradients in the three image dimensions. Thus, inhomogeneities in the local field can influence the imaging gradients leading to a miscalculated field of view (FOV) and spatial distortion. The amount of distortion is proportional to the field inhomogeneity and the data acquisition time. Echo Planar Imaging (EPI) sequences are especially susceptible to inhomogeneity induced distortion due to the acquisition of a whole slice of raw data after the excitation pulse. This leads to strong distortions in the phase encoding direction, while orthogonal directions have negligible distortion due to the short difference in acquisition time. Thus, distortion correction involves repositioning voxels in only one dimension.

The the displacement of a given voxel in the phase encoding direction can be found as follows using a fieldmap:

$$d\mathrm{PE}(x, y, z) = \gamma \Delta B_0(x, y, z) T\mathrm{ro} \tag{2.1}$$

Where $d\mathrm{PE}(x, y, z)$ is the displacement of the voxel along the phase encoding direction, γ is the gyromagnetic ratio $\Delta B_0(x, y, z)$ is the fieldmap and Tro is the slice readout time (Hutton et al. 2002). The fieldmap is calculated from the output of the GRE fieldmap sequence as follows:

$$\Delta B_0 = \frac{\Delta\Theta}{2\pi\gamma\Delta\mathrm{TE}} \tag{2.2}$$

Where $\Delta\Theta$ is the phase difference between the different TEs, γ is the gyromagnetic ratio, and ΔTE is the difference in the echo times (Hutton et al. 2002). Following this, the voxels in the anatomical image and the calculated reference EPI are corrected for the inhomogeneity induced voxel displacement.

2.2.5 Denoising neural data

Subjects with a maximum framewise displacement higher than 2mm, mean framewise displacement higher than 1mm and having more than 2TRs with framewise displacement higher than 1mm in the whole duration of the experiment were removed. Following this a 6mm isotropic full-width half-maximum (**FWHM**) was applied to smooth the preprocessed EPI volumes. To account for additional sources of noise, various confounds calculated by the preprocessing pipeline were regressed from the data. The regressors include the motion parameters along which the affine transformations were made to correct the images, activity associated with CSF and WM calculated using the CompCore method, framewise displacement values, and cosine regressors to account for scanner drift. All denoising was performed using NiLearn (contributors n.d. ,RRID:SCR-001362)

2.2.6 Spearman correlation analysis

As our sample isn't large enough to assume normality, we have preferentially used Spearman's rank correlation as the primary correlation coefficient rather than Pearson. Spearman correlation is a non-parametric equivalent of the Pearson correlation coefficient. This coefficient is measured by assigning ranks to the observations of the variables under study, and calculating the Pearson correlation of the ranks (Equation 2.3). While significance of the correlation is calculated as a t-statistic using equation 2.5. While a Pearson coefficient tests for a linear relationship between variables, a Spearman correlation assesses a monotonic relationship (Spearman 1904).

$$\rho \mathrm{s} = \frac{\mathbf{cov}[\mathrm{R}[X], \mathrm{R}[Y]]}{\sigma \mathrm{R}[X], \sigma \mathrm{R}[Y]}, \tag{2.3}$$

$$\text{and } \boldsymbol{t} = \rho \mathrm{s} \sqrt{\frac{n-2}{1-\rho^2}}. \tag{2.4}$$

We performed this using the spearmanr tool within the SciPy package (Virtanen et al. 2020). As this package has low accuracy for calculation of the t-statistic for significance testing at N<300, we utilised a permutation test (independent sorting, 10000 shuffles) for hypothesis testing.

2.2.7 General Linear Models for finding neural correlates

The General Linear Model is an extension of a basic linear regression model that additionally allows you to test the predictive capacity of multiple predictors. It encompases multiple linear regression equations into a single relationship that can be easily computed. A GLM equation can be written as follows,

$$Y\mathrm{i} = \beta_0 + \beta_1 \mathrm{X}_1 + \cdots + \beta \mathrm{iXi} + \epsilon \mathrm{i} \tag{2.5}$$

$$\text{or } Y = \beta_0 + \sum^{N} \mathrm{i} = 1 \beta \mathrm{iXi} + \epsilon \tag{2.6}$$

Following this the p-value was corrected for multiple comparisons, controlling the false discovery rate (FDR) at $\alpha = 0.05$, using the Benjamini-Hochberg correction. Furthermore, a cluster threshold of 10 voxels was also applied, i.e any cluster will have to be at least 10 voxels in size to appear in the final 2nd-Level result. A cluster threshold helps remove stray voxels that appear to be involved.

Chapter 3

Results

The stimulus was constructed using ragas characterised within Mathur et al. 2015. They employed discrete rating of the individual ragas for distinct emotions. This aided in the construction of an expected response curve for the valence of the stimulus. The experimental data was found to be highly correlated to this expected valence curve as seen in ***Fig:*** 3.1 (Tripathi et al. 2025). These findings strongly validate the stimulus and the paradigm's ability to effectively capture emotional fluctuations. This strengthened our choice in utilising this dataset to pursue our hypothesis.

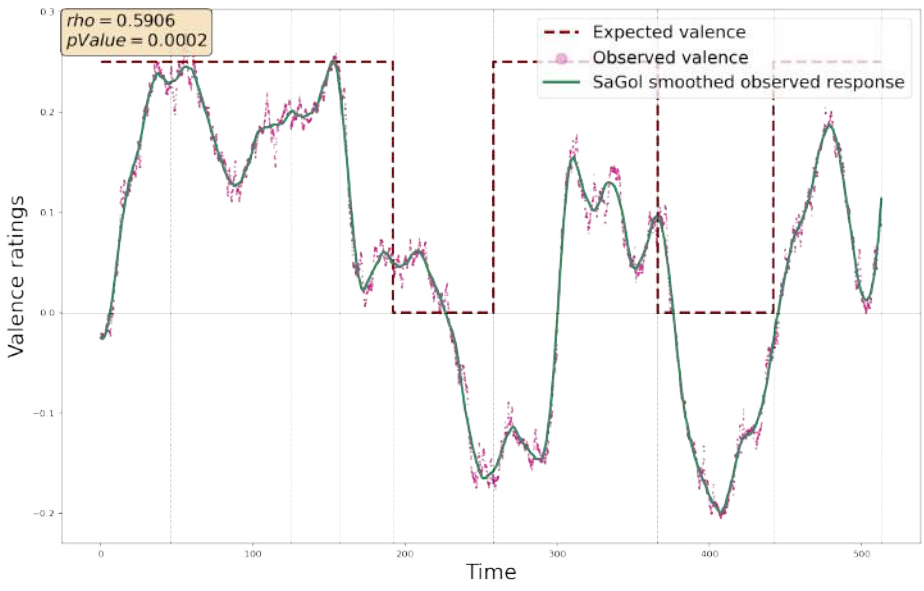


Figure 3.1: Group level responses strongly resemble (rho ≅ 0.59, p ≅ 0) the valence values proposed in Mathur et al. 2015, validating stimulus strength and paradigm design, Recreated from: Tripathi et al. 2025.

3.1 Using kinematics to glimpse into the emotional brain

The emotional music dataset contains the position of every emotional report by an individual on a two-dimensional affective space (Tripathi et al. 2025). This information can be interpreted as a spatial representation of the path taken by the brain to generate an emotional response to the stimulus. Thus, we hypothesised that approaching this information through the lens of physical kinematics would allow us to study higher order representations of affective information.

Specifically, it would allow us to study the variation of both valence and emotional arousal in a unified fashion, this would not be possible with traditional experimental approaches. We calculated various kinematic features of the participant behaviour using a central difference scheme. They were calculated using the following formulae,

$$\overrightarrow{\chi_t} = \sqrt{(\overrightarrow{V}_{t+1} - \overrightarrow{V}_{t-1})^2 + (\overrightarrow{A}_{t+1} - \overrightarrow{A}_{t-1})^2} \quad , \tag{3.1}$$

$$\overrightarrow{v_t} = \frac{\chi_t}{2\Delta \mathrm{T}} \quad , \tag{3.2}$$

$$\text{and } \overrightarrow{a_t} = \frac{v_{t+1} - v_{t-1}}{2\Delta \mathrm{T}} \quad . \tag{3.3}$$

In (3.1), $\overrightarrow{A}$ and $\overrightarrow{V}$ are the emotional arousal and valence respectively, obtained from Tripathi et al. 2025. In (3.2), $\overrightarrow{\chi_t}$ is the displacement of the joystick, while in (3.3), $\overrightarrow{a_t}$ is the acceleration. Δ T is the size of the sampling window, or the difference between two consecutive time points.

With the data for each kinematic feature at hand, we wanted to utilise them to capture stable and transitory emotional states. As acceleration (3.3) is a second order derivative, it can capture the rate of change, of the rate of change of emotional state of the participant (*position*). Thus, the acceleration time series would have distinct markings of when a participant is moving (transitory phase) and when they are not (stable phase). We then binarised the acceleration time series, to produce a raster plot of fluctuations in the affective state of all individuals.

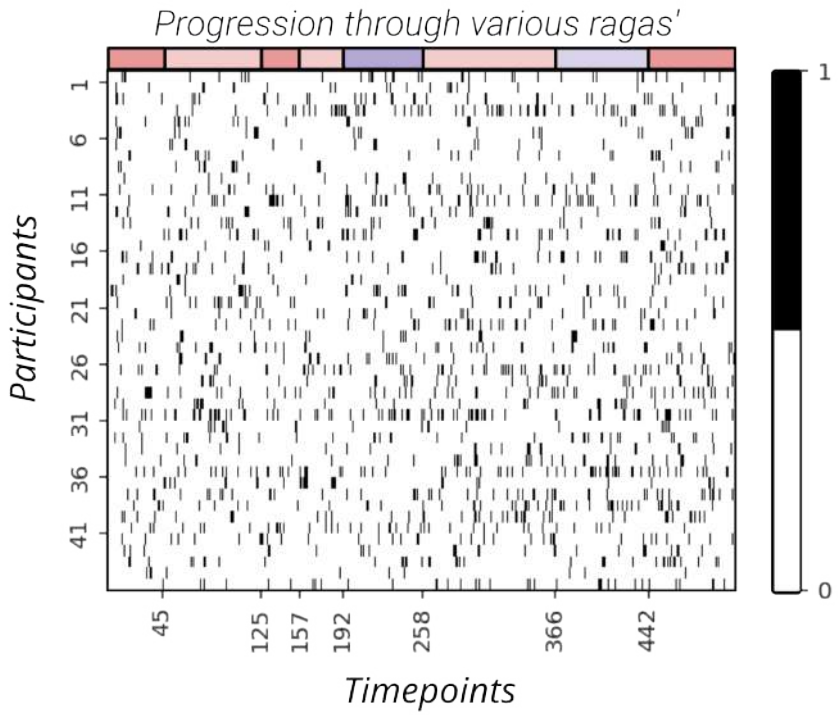


Figure 3.2: Raster plot of emotional transitions in an individual derived by binarisation of the acceleration time series

Every non-transition or stable (0) state represents time spent in a stable and defined emotional space, and every transitioning (1) state represents timepoints where the individual is transitioning across emotional states (***Fig:*** 3.2). Since each transition event represents an emotional shift in the participant, we hypothesised that the cumulative number of shifts should be associated with their emotionality scores.

To test this hypothesis, we calculated two measures of emotionality, the number of shifts derived from the kinematic raster plot, and the other being the the span of their emotional path on the tw-dimensional affective space. Theoretically, these measures should not differ significantly and should be associated to the emotionality of the participants.

$$S = \frac{\sum \sqrt{\overrightarrow{V}^2 + \overrightarrow{A}^2}}{N} \tag{3.4}$$

In (3.4), S is the emotional span, $\overrightarrow{V}$ and $\overrightarrow{A}$ are valence and emotional arousal obtained from Tripathi et al. 2025, and N is the total number of timepoints.

A rising trend with emotionality scores was observed in both measures, confirming that both are indeed capturing the emotionality of the participant (3.3 and 3.4). Although we observed that cumulative number of shifts had a stronger correlation and a leaner confidence interval along the regression line (3.4), implying a stronger association with emotionality.

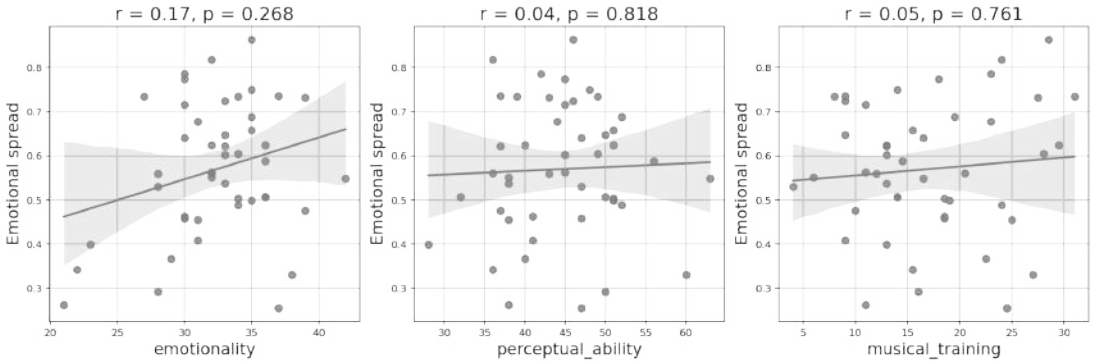


Figure 3.3: Correlation between emotional spread and trait scores

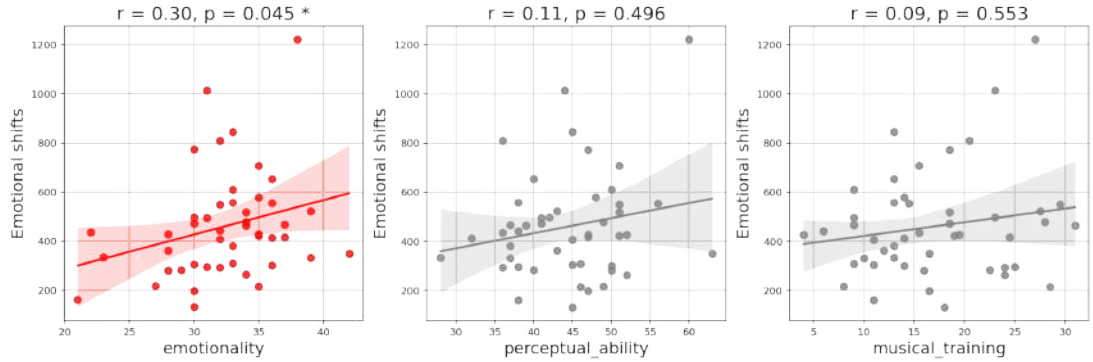


Figure 3.4: Correlation between number of shifts and trait scores

3.2 Deriving emotional reactivity from kinematics

Having established kinematic features to be a valid representation of emotional behaviour, we wanted to explore how these features could be utilised to investigate affective flexibility. Thus, we set out to derive a measure of emotional reactivity from these features. Emotional reactivity is a measure of how fast an individual can produce a emotional response to changes in their environment. This makes it a key attribute of affective flexibility, as it is a measure of how quickly an individual can perceive a transition in the affective value of their environment and respond with a corresponding change in their state.

3.2.1 Latency of response as a proxy for emotional reactivity

Emotional reactivity, in the context of this study, is defined as the amount of time taken from an event boundary in the stimulus to the report of a shift in their emotional state (*latency of response*). Since we knew clearly where the boundaries in the stimulus lie, the time till the first acceleration event after the shift in stimulus would be the latency of response or *emotional reactivity*.

Through the behavioural raster we had previously derived (***Fig:*** 3.2), we searched for the closest report of a change in emotional state i.e the time till the first transition event after the stimulus changes, as shown in ***Fig:*** 3.5.

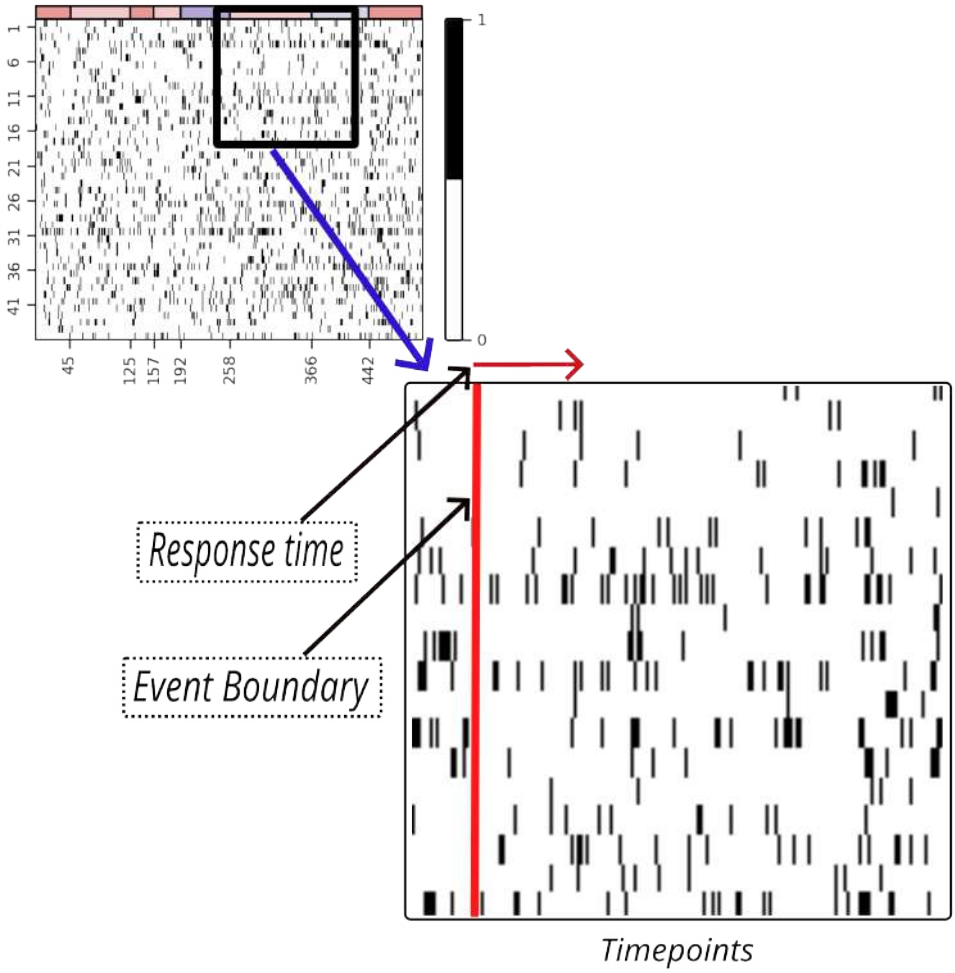


Figure 3.5: Deriving emotional response time from kinematic features

The response latencies obtained were normalised for event duration and then Z-score standardised within participants to reduce the effect of individual differences in rating behaviours and make them comparable. As our sample population is made up of healthy individuals we expected that there won't be any difference in the median response latencies across the different ragas. Surprisingly, a Friedman chi square test pointed towards differences between the latency medians (p-value $\approxeq$ 0.0004). Unfortunately, these differences didn't survive in a post-hoc Wilcoxon signed-rank test (Holm-corrected p-value $\approxeq$ 1 for all comparisons), indicating some variability in the response latencies.

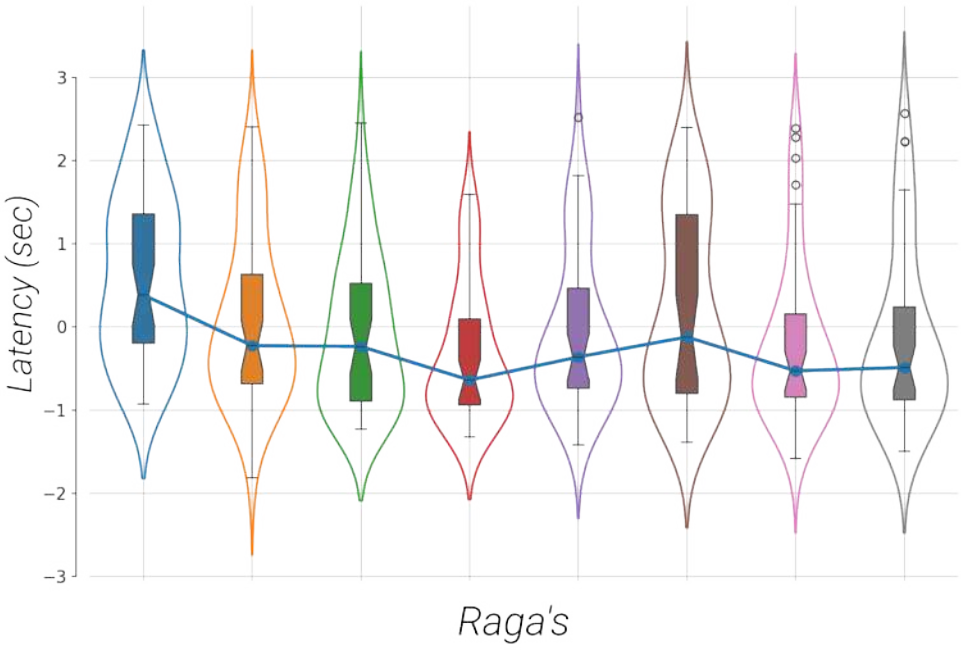
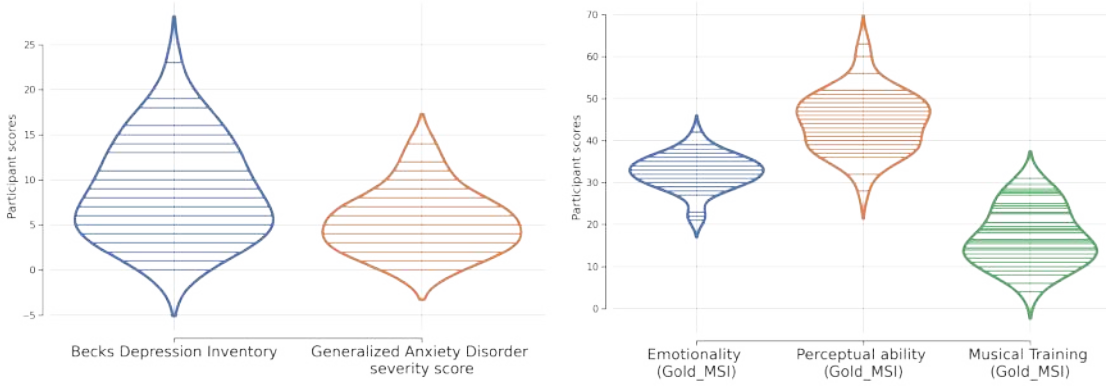


Figure 3.6: Response latencies across various ragas

3.3 Kinematics reflect trait induced individual variability

We wanted to explore what aspects of individual personality and traits contribute to variability in affective behaviours. We employed questionnaires as an independent measure of personality and traits. We used the Gold-MSI questionnaire scores as a measure of individual ability to discern emotional content of the music (*emotionality*), *perceptual abilities* and *musical training*. We also used the Beck's Depression Inventory (*BDI*) and General Anxiety Disorder severity scale (*GAD* − 7) as a measure of depressive and anxious traits in all participants.



(a) Neuropsychiatric scores of the participants

(b) Musicality scores of the participants

Figure 3.7: Distribution of trait scores for all participants used within this study

All individuals in our sample had subclinical neuropsychiatric scores, thus, their behaviours shouldn't be significantly different affected by these traits. Hence, we hypothesised that majority of the variability in emotional reactivity would be captured by attributes of the Goldsmith-MSI questionnaire.

3.3.1 Median emotional reactivity is influenced by trait features

While statistically insignificant, we observed that emotionality had a weak positive correlation, while both perceptual abilities and musical training had a weak negative correlation (***Fig:*** 3.8). Furthermore, as expected, there is no significant impact of both subclinical depressive and anxious behaviours on the median emotional reactivity (***Fig:*** 3.9). Although, the wide confidence intervals imply the presence of an association that cannot be significantly established with the current sample size.

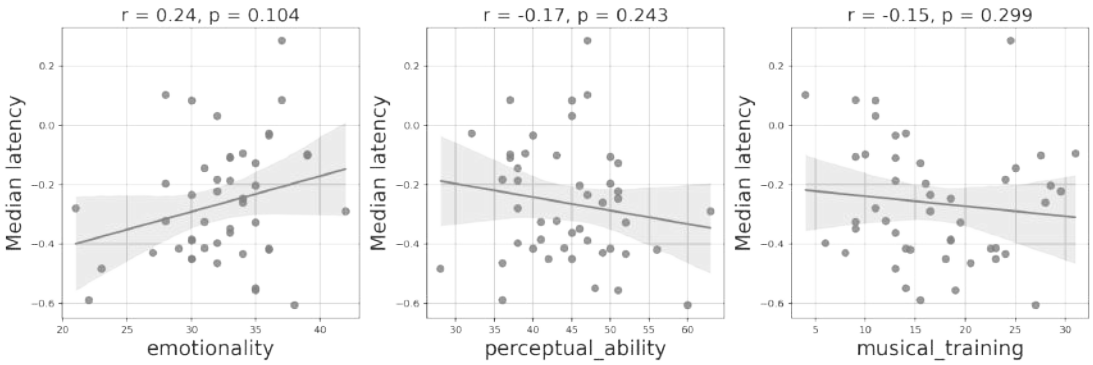


Figure 3.8: Emotionality potentially impacts emotional reactivity

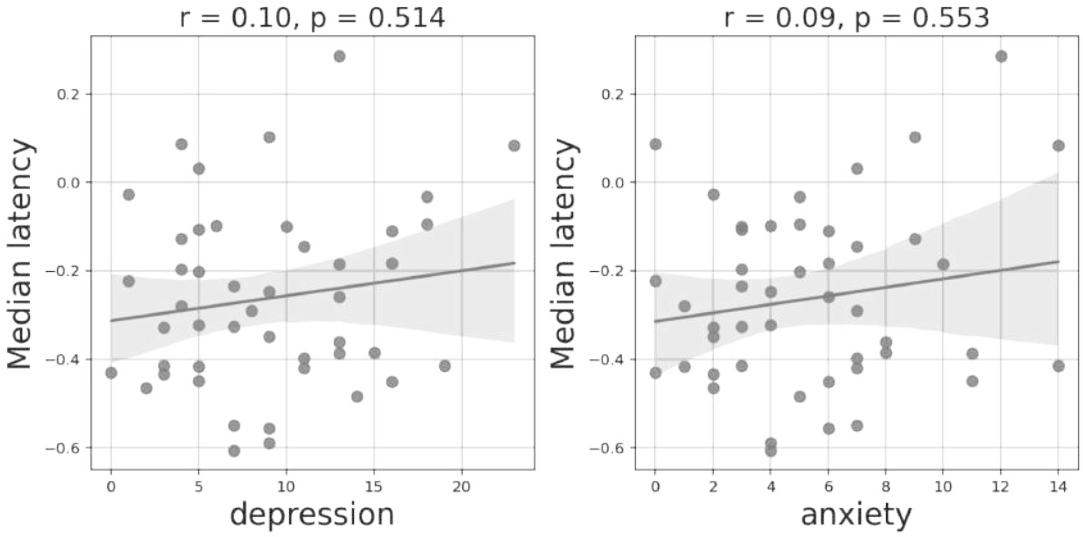


Figure 3.9: Median emotional reactivity is not affected by subclinical depressive and anxious behaviours

3.3.2 Presence of neuropsychiatric traits negatively impacts emotional reactivity

Having found no significant impact of traits on median emotional reactivity, we next explored their impact on transitions within a dimension. We limited our exploration to the valence dimension as we had prior characterization of the valence of the emotional content in the stimulus (Mathur et al. 2015, ***Fig:*** 3.1).

The stimulus contains two distinct forms of emotional shift, from positive to negative valence and vice-versa. There is a positive to negative valence shift in the stimulus when the raga changes from *Rageshree* to *Basant mukhari*, and from *Tilak kamod* to *Marwa*. While a negative to positive valence shift happens when the raga changes from *Basant mukhari* to *Tilak kamod*, and from *Marwa* to *Jog* (***Fig:*** 3.1).

We had hypothesised a negative correlation between emotional reactivity and neuropsychiatric scores for a positive to negative shift, displaying a tendency to move faster towards a negative valence state. Surprisingly, we found these shifts were not at all affected by neuropsychiatric scores (***Fig:*** 3.10). The absence of a trend in the regression line further implied the absence of an association.

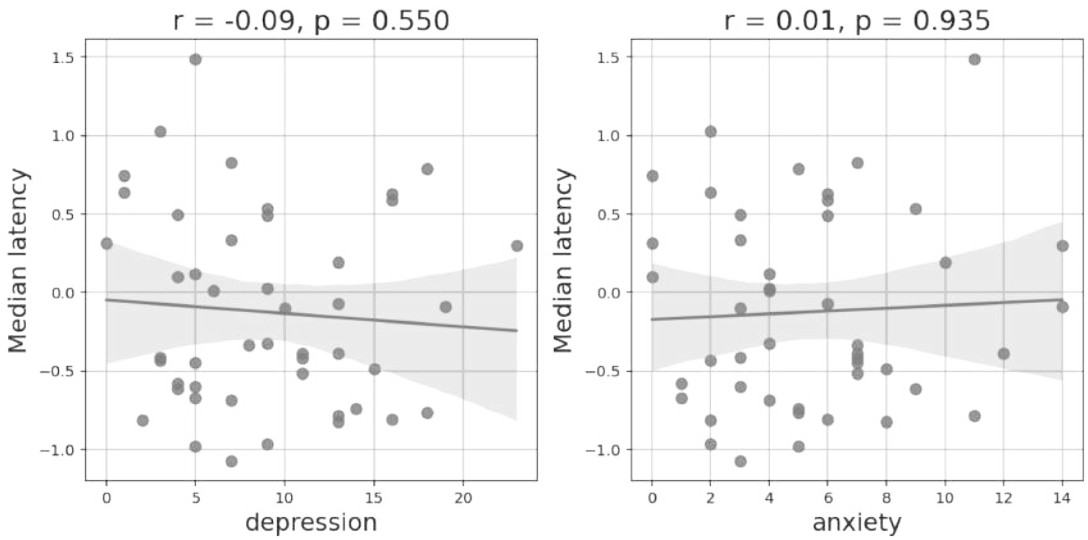


Figure 3.10: Subclinical neuropsychiatric behavioural traits don't influence emotional reactivity in positive to negative valence transitions

However, we discovered that there is a stark rise in the tendency to stay within the negative valence state, when the stimulus shifts towards a positive valence, with increase in neuropsychiatric scores (***Fig:*** 3.11). This hints at an association between an increased difficulty in transitioning in a specific direction and depressive or anxious

traits. This association is further supported by the lean confidence intervals around the regression line, indicating a strong relationship.

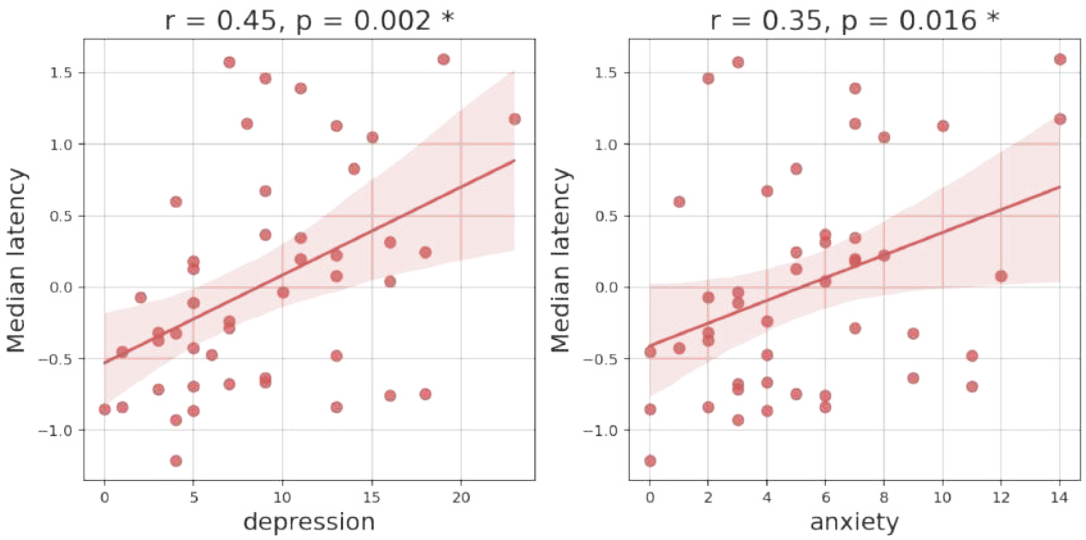


Figure 3.11: Depressive and anxious behaviours significantly affect emotional reactivity when transitioning away from negative valence states

While we hadn't obtained statistically significant relationships between median latency and the MSI trait scores, we expected similar trends to appear when transitions across specific directions are studied. Surprisingly, we observed that in either directions of transition, trait scores had little to no influence on the latency of response (***Fig:*** 3.12 and ***Fig:*** 3.13).

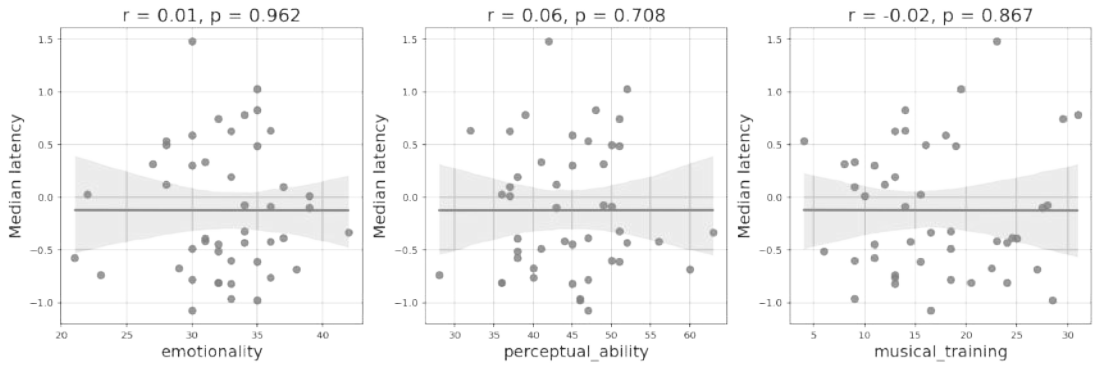


Figure 3.12: Trait features have no influence in + to - valence transitions

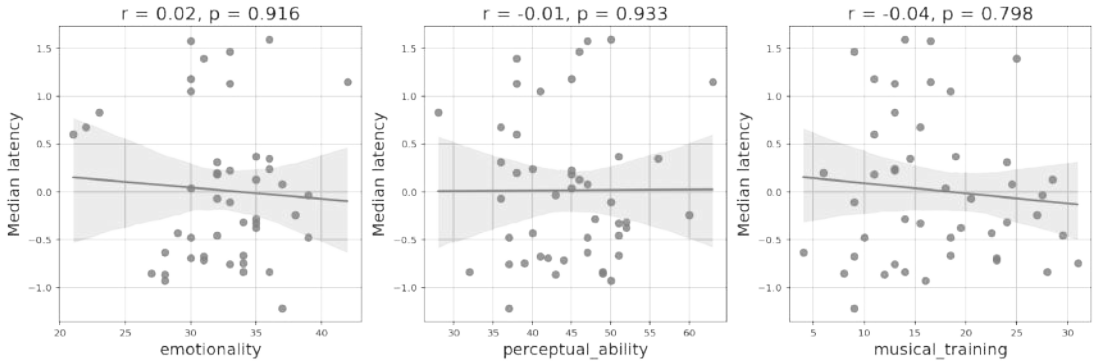


Figure 3.13: Trait features have no influence in - to + valence transitions

3.4 Neural data validates affective features captured by kinematics

As neuroscientists the final frontier for all of our works is the ability to showcase the phenomena we study within the brain.

While the behavioural results strongly captured the validity of kinematic features as a measure of affective flexibility and attributes of emotional behaviour, we wanted to further validate these findings through neural data. We aimed to find the neural regions involved in emotional transitions and investigate how their activity culminated in an evolving emotional experience. We again employed the raster plot (***Fig:*** 3.2) we had previously generated, as a guiding marker for finding regions involved in these tasks.

We built a Generalised Linear Model to find regions specifically activated within the acceleration and non-acceleration events, corresponding to regions active in stable emotional experience and regions involved in transitioning across emotional states. A 1st-Level model was setup to find regions at an individual level, followed by a 2nd-Level model to find activations common in the group (***Fig:*** 3.14). The design matrix (***Fig:*** 3.14a) was constructed by convolving the Glover HRF model (Glover 1999) with the simulated activation time-series for each condition (Acceleration (A) and No-Acceleration (NA)). We chose to employ the Glover HRF model as it is a widely accepted model of the haemodynamic response function (HRF) in functional MRI studies, and because the original study was conducted using an auditory task.

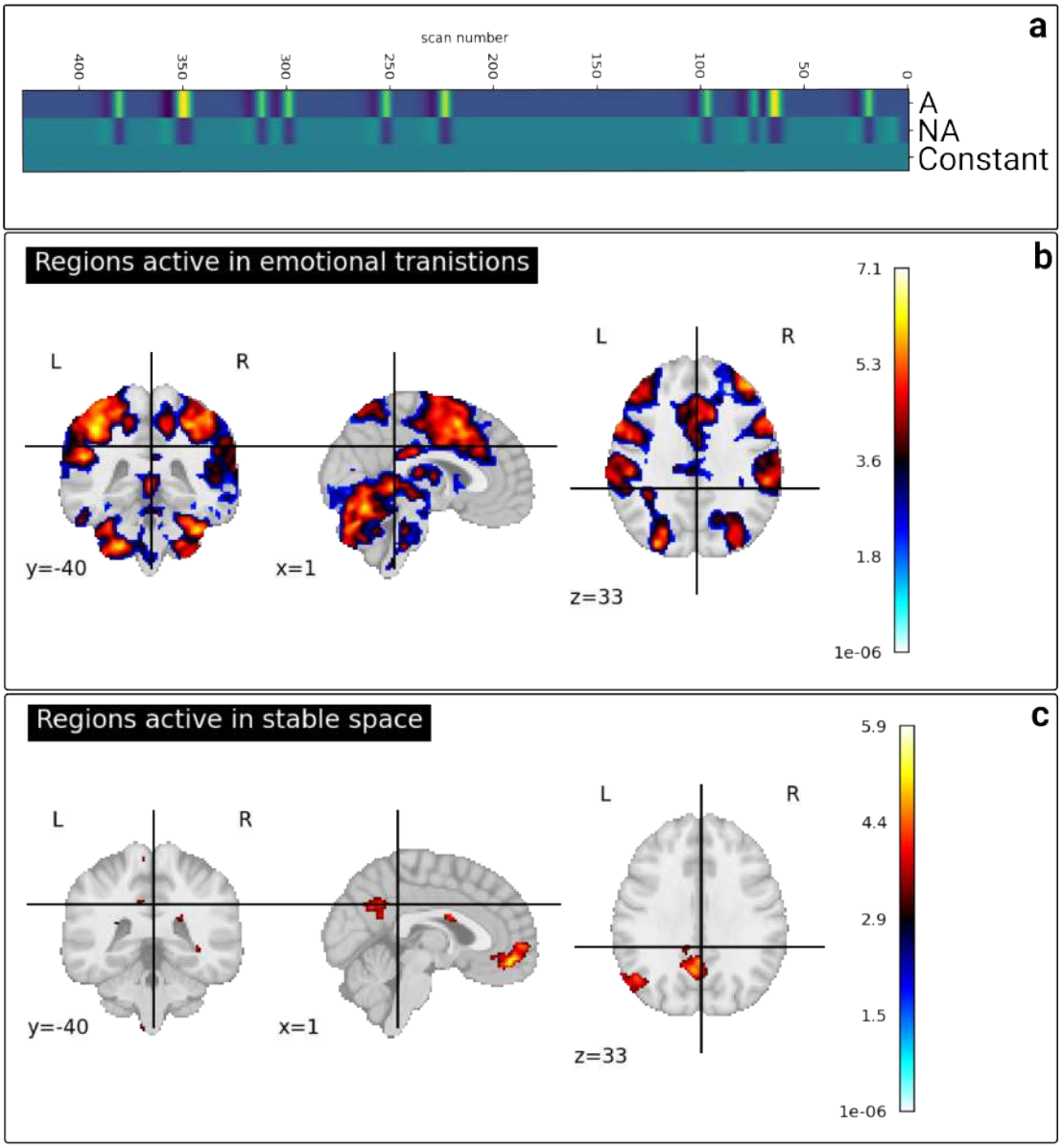


Figure 3.14: Z scored stat maps obtained from the 2nd level GLM

While many regions appeared in the contrasts, post-hoc correction to control for false positive reports of activations greatly reduced the regions that were observed. The orbitofrontal cortex and the parietal cingulate were the two most prominently active regions in the stable state contrast (NA-A). The transitory state contrast (A-NA) had a wide range of regions appearing to be prominently active, these include, the intra-parietal sulcus and lobule (IPS and IPL), parts of the anterior cingulate cortex (ACC), pars opercularis, and the anterior insula.

Chapter 4

Discussion

The primary goal of our study was to develop a measure of affective flexibility and explore individual variability through this measure. With their motor behaviour as a proxy for their emotional behaviour, we wanted to understand how individuals adapt their emotional responses to changing situations. We had predicted that the kinematics features of their motor behaviour would be able to predict the extent of affective flexibility in an individual.

Acceleration as a measure of flux in the emotional space

A continuous capture paradigm provided us with the path traced by the participants in their individual emotional space. At face value, the path only provided us with an idea of the absolute emotional experience of the individual, similar to how positional information of a moving object only provides us with its location. However, the path also provided us with the opportunity to extract higher kinematics features, such as the rate of change of their emotional experience (velocity 3.2) and the rate of change of this rate of change (acceleration 3.3).

As we had predicted, we found that the acceleration events were indeed representative of stable and transitory emotional states (3.14) capturing emotional and control networks within the brain. The cumulative shift in emotional space, a measure derived from the acceleration events, was found to have a significantly stronger representation of individual emotionality than the span of their emotional path (area of the emotional space traversed). This is potentially because the span of the emotional path is a noisy measure that only accounts for the area of the emotional space traversed by the individual, making it susceptible to under-representation of individuals with smaller spans but more frequent shifts. Meanwhile, the cumulative number of shifts captures the total number transitions in the emotional space of an individual, this allows it to better capture the emotionality of an individual who has a smaller span but more frequent

shifts.

Affective flexibility and individual variability

Having established a measure of emotional flux, we were also able to establish a strong measure of affective flexibility 3.5. Through literature survey, we had identified that emotional regulation and thus affective flexibility have a strong temporal component. Thus, we had predicted that changes in the stimuli would be reflected in the emotional state of the participant with a certain delay. Furthermore, we had predicted that this delay would vary depending on the nature of the change in the stimuli, as varying number of cycles of attentional appraisal would have to be implemented to evaluate and generate a response. Hence, theoretically every change in the raga would have a delay in response characteristic to it, with minor differences across the participants.

While we did discover that the median response latencies differed across the ragas, none of the differences passed a p-corrected post-hoc analysis 3.6. This captured the minor variations in responses we predicted formerly. This is a reflection of the minor differences in the perception of the stimuli across the participants.

Influence of traits on individual variability in affective flexibility

We had predicted that individual variability would be strongly influenced by the personality traits of the participants. Thus, it was surprising to discover that traits had no significant association with median latency. We hypothesised that this could be due to the relatively smaller sample size, and the participants being subclinical in their neuropsychiatric traits.

Focusing on transitions across specific dimensions brought out the influence of traits and made individual variability more apparent. When the stimulus transitions across a specific dimension, there is a stark and obvious change in the emotional content of the environment. This would require repeated cycles of attentional appraisal that would be reflected in the response latency. Additionally, in individuals with higher neuropsychiatric scores, like BDI, there would be a greater delay due to the presence of conflicting internal processes and flawed regulatory strategies that interfere with the appraisal process (Genet et al. 2013; Malooly et al. 2013).

As expected, we found that the response latencies were significantly longer for transitions from negative to positive valence.

The absence of an association being observed in the transition from positive to negative valence indicated the absence of a preference for negatively valent stimuli, which is consistent with the fact that the participants were subclinical in their neuropsychiatric traits. We did find that a higher BDI score significantly impacted affective

flexibility, with longer response latencies being observed. This is consistent with our prediction that a higher BDI score would lead to rumination like coping behaviours that would interfere with valuation of changes in the environment, decreasing affective flexibility.

Limitations and future perspectives

The strongest limitation of our study was the relatively small sample size, which severely impacted the statistical power of our analysis. With a larger sample size we expect that a stronger association would be observed between traits and affective flexibility. Another limitation is the absence of clinical samples - the influence of neuropsychiatric traits we have captured would be better represented in a clinical sample. Furthermore, the presence of a clinical sample would allow us to investigate mechanistic differences that are otherwise masked by the subclinical nature of our sample group.

Having established a strong measure of both affective flexibility and emotional flux, delving deeper into the neural mechanisms underlying these processes would be the logical next step. As emotions have been established to be a product of the interaction of multiple networks within the brain (Pessoa 2017), we believe the trait dependent variability we observe in the behaviour would be strongly reflected in the network dynamics too.

Finally, we would like to explore the potential of using this measure of affective flexibility in clinical settings. This measure would allow for an unbiased assessment of the subject without the need for self-report questionnaires, which can be biased by the subject's own perception of their mental and emotional state.

Appendix A

Code listing

Functions used for processing behavioural data

```
function behavProcess(myFolder, pattern, destFin, timeprd, incmt,
    rescaleData, deadzone, normalizeData, dzFac, val_ind, arl_ind,
    RangeMax, RangeMin, InputMax, InputMin)
% The following code takes in raw participant behavioural data
    from
% specified location (myFolder), downsamples it to specified
    timeperiod
% (timeprd), with specified increments(incmts). Then it rescales
    the
% downsampled participant data to a specified range (RangeMax,
% RangeMin) from the inherent input range (InputMax, InputMin). If
     the
% deadzone variable is true it then applies a deadzone at the
    center of
% a specified threshold (dzFac). Finally it writes the processed
    data
% to a file named sub-%%_processed.csv

% Read in the file location details
filePattern = fullfile(myFolder, pattern); % Change to whatever
    pattern you need.
theFiles = dir(filePattern);

% Initialization of variables
timescale = (0:incmt:timeprd)';
val_data = zeros(length(timescale),length(theFiles));
arl_data = zeros(length(timescale),length(theFiles));
```

```
    normVal = zeros(length(timescale),length(theFiles));
    normArl = zeros(length(timescale),length(theFiles));
    valData = zeros(length(timescale),length(theFiles));
    arlData = zeros(length(timescale),length(theFiles));

    % DOWNSAMPLING
    % Loop over file list and read in the behaviour data
    for k = 1 : length(theFiles)
        baseFileName = theFiles(k).name;
        fullFileName = fullfile(theFiles(k).folder, baseFileName);
        fprintf(1, 'Now reading %s\n', baseFileName);

        % Read in the file
        emo_data = readmatrix(fullFileName);

        %Loop over the num of observations in the final table

        for i = 1:length(timescale)
            % Timescale is a incmt multiple of every observation num
            cond = round((i-1)*incmt,2);
            % Check which timepoints come close to the established
                timepoint (cond)
            [minVal,minInd] = min(abs(emo_data(:,2)-cond));
            % Index out closest timepoint from the original data
            val_data(i,k) = emo_data(minInd,val_ind);
            arl_data(i,k) = emo_data(minInd,arl_ind);
        end
    end

    % RESCALING
    % Rescale the valence and arousal datapoints to specified range
    if rescaleData == true
        for k = 1 : length(theFiles)
            % Rescales the data to specified range
            valData(:,k) = rescale(val_data(:,k),RangeMin,RangeMax,"
                InputMax",InputMax,"InputMin",InputMin);
            arlData(:,k) = rescale(arl_data(:,k),RangeMin,RangeMax,'
                InputMax',InputMax,'InputMin',InputMin);
        end
    else
        valData = val_data;
```

```
57          arlData = arl_data;
58      end
59
60      % DEADZONING
61      % Apply deadzone and center participant data across participants
62      % within time points
63      if deadzone == true
64          for k = 1 : length(theFiles)
65              for rowInd = 1:length(timescale)
66                  % Anything between +/- deadzone filter is made zero
                        for
67                  % both valence and arousal
68                  if valData(rowInd,k) <= dzFac && valData(rowInd,k) >=
                        -dzFac
69                     valData(rowInd,k) = 0;
70                  end
71                  if arlData(rowInd,k) <= dzFac && arlData(rowInd,k) >=
                        -dzFac
72                     arlData(rowInd,k) = 0;
73                  end
74              end
75          end
76      end
77
78      % NORMALIZATION
79      if normalizeData == true
80          % Normalize participant responses
81          valData = normalize(valData, 1,"zscore");
82          arlData = normalize(arlData, 1,"zscore");
83      end
84
85      % SAVING FINAL DATA
86      % Output the final data and save it to specified file format
87      for k = 1 : length(theFiles)
88          baseFileName = theFiles(k).name;
89          name = split(baseFileName,"_");
90          fprintf(1, 'Now preparing to write processed data for subject
                %s\n', name{1});
91          final_table = zeros(length(timescale),3);
92          final_table(:,1) = timescale; % Timescale created by us
93          final_table(:,2) = valData(:,k); % Initial 5 time points are
```

```
            zero
94          final_table(:,3) = arlData(:,k); % Initial 5 time points are
                zero
95          %final_table(1,:) = zeros(1,width(final_table));
96          resultname = strcat(destFin, name{1},'_processed.csv');
97          writematrix(final_table,resultname,"WriteMode","overwrite");
98      end
99  end
100
101
102 function calcKinematics(myFolder,pattern, val_ind, arl_in,time_ind,
        strength_ind, disp_ind, vel_ind, acc_ind, jerk_ind, filter, fltVal)
103 % The function takes in cleaned and pre-processed behaviour time
        series and
104 % calculates the kinematics for that participant
105
106 % myFolder specifies the path to be checked for sources files, pattern
107 % tells the function the name pattern of the files to be looked for
108 % val_ind, arl_in, time_ind specify the indices for valence, arousal
        and
109 % the time in the source files. Displacement, velocity and
        acceleration
110 % indices specify the indices where the corresponding kinematics
        values are
111 % to filled in.
112
113 % Filter specifies whether the kinematics should be thresholded on the
114 % basis of displacement distribution for the participant. fltVal
        specifies
115 % the number of standard deviations the kinematics distribution should
         be
116 % filtered within.
117
118     % Get a list of all files in the folder with the desired file name
            pattern.
119     filePattern = fullfile(myFolder, pattern); % Change to whatever
            pattern you need.
120     theFiles = dir(filePattern);
121
122     % Loop through the file list
123     for k = 1 : length(theFiles)
```

```
124         baseFileName = theFiles(k).name;
125         fullFileName = fullfile(theFiles(k).folder, baseFileName);
126         fprintf(1, 'Now reading %s\n', fullFileName);
127 
128         % Read in the participant data
129         emo_data = readmatrix(fullFileName);
130 
131         % Initialize empty rows and fill them with zeros, to be filled
                later
132         emo_data(1,disp_ind) = zeros();% Displacement
133         emo_data(1,vel_ind) = zeros();% Velocity
134         emo_data(1,acc_ind) = zeros();% Acceleration
135 
136         fprintf(1, 'Calculating resultant displacement, velocity and
               accelaration \n')
137 
138         % Indexes from 2 because Intensity at t = 0/1 will be 0
139         for i = 1:length(emo_data)
140             % Calculate the intensity of emotion at each
141             % consecutive time frame.
142             % Intensity  = sqrt (valence(t)^2 + arousal(t)^2)
143             emo_data(i,strength_ind) = sqrt( (emo_data(i,val_ind) +
                    emo_data(i,arl_in) )^2 );
144         end
145 
146         % Indexes from 2 because Displacement at t = 0/1 will be 0
147         for i = 2:length(emo_data)-1
148             % Calculate the displacement of the joystick at each
149             % consecutive time frame.
150             % Displacement = sqrt( (valence(t) - valence(t-1))^2 + (
                    arousal(t) - arousal(t-1))^2)
151             emo_data(i, disp_ind) = sqrt( (emo_data(i+1, val_ind) -
                    emo_data(i-1, val_ind) )^2 +  (emo_data(i+1, arl_in) -
                    emo_data(i-1, arl_in) )^2 );
152         end
153 
154         if filter == true
155             % Calculate the standard deviation of the displacement
                    values
156             % for the participant. This will be used as a filter and
                    only
```

```
157             % displacements "above" fltVal*SD will be allowed to be
                    used
158             % for velocity calculation.
159             sdThresh = fltVal * std(emo_data(:,disp_ind));
160             % Indexes from 2 because velocity at t = 0/1 will be 0
161             for i = 3:length(emo_data)-1
162                 if emo_data(i,disp_ind) <= sdThresh && emo_data(i,
                        disp_ind) >= -sdThresh
163                     % Make any timepoint with displacement values less
                            than
164                     % sdThresh zero
165                     emo_data(i,disp_ind) = 0;
166                     %emo_data(i,strength_ind) = 0;
167                 else
168                     % Calculate the velocity of the joystick at each
169                     % consecutive time frame.
170                     % Velocity = displacement(t-1:t)/time(t-(t-1))
171                     emo_data(i, vel_ind) = emo_data(i, disp_ind) /
                            2*((emo_data(i, time_ind) - emo_data(i-1,
                            time_ind)));
172                 end
173             end
174         else
175             for i = 3:length(emo_data)-1
176             % If filter is not true velocity is calculated
                    unthresholded
177             % Velocity = displacement(t-1:t)/time(t-(t-1))
178                 emo_data(i, vel_ind) = emo_data(i, disp_ind) / 2*((
                        emo_data(i, time_ind) - emo_data(i-1, time_ind)));
179             end
180         end
181
182         % Indexes from 4 because Acceleration at initial two time
                points will be 0
183         for i = 4:length(emo_data)-1
184             % Calculate the acceleration
185             % Acceleration = (velocity(t) - velocity(t-1))/time(t-(t
                    -1))
186             emo_data(i,acc_ind) = (emo_data(i+1, vel_ind) - emo_data(i
                    -1, vel_ind)) / 2*(emo_data(i, time_ind) - emo_data(i
                    -1, time_ind));
```

```
187         end
188
189
190         for i = 5:length(emo_data)-1
191             % Calculate the acceleration
192             % Acceleration = (velocity(t) - velocity(t-1))/time(t-(t
                    -1))
193             emo_data(i,jerk_ind) = (emo_data(i+1, acc_ind)-emo_data(i
                    -1, acc_ind)) / 2*(emo_data(i,time_ind)-emo_data(i-1,
                    time_ind));
194         end
195         % Write the values to the specified file
196         writematrix(emo_data, fullFileName,'WriteMode','overwrite');
197
198     end
199 end
200
201 function [valence, arousal, strength, displacement, time, velocity,
        acceleration, jerk] = disFeat_Exact(myFolder,pattern,val_ind,
        arl_ind, strength_ind, disp_ind, time_ind, vel_ind,acc_ind,
        jerk_ind)
202
203 % This function has been create to extract state and kinematics
        information
204 % from participant files with ease without having to call out each
205 % individual participant repeatedly.
206
207     % Get a list of all files in the folder with the desired file name
            pattern.
208     filePattern = fullfile(myFolder, pattern); % Change to whatever
            pattern you need.
209     % Contains the names/locations of files with specified name
            pattern
210     theFiles = dir(filePattern);
211
212     % Initialise empty vectors of required length
213     valence = [1,length(theFiles)];
214     arousal = [1,length(theFiles)];
215     strength = [1,length(theFiles)];
216     displacement = [1,length(theFiles)];
217     velocity = [1,length(theFiles)];
```

```
218     time = [1,length(theFiles)];
219     acceleration = [1,length(theFiles)];
220     jerk = [1,length(theFiles)];
221 
222     % Loop across files/participant numbers
223     for k = 1 : length(theFiles)
224         baseFileName = theFiles(k).name;
225         fullFileName = fullfile(theFiles(k).folder, baseFileName);
226         fprintf(1, 'Now reading %s\n', fullFileName);
227 
228         % Load participant data files
229         emo_fdata = readmatrix(fullFileName);
230 
231         % Look across each time points
232         for i = 1: length(emo_fdata)
233             % Indexing out each variable and filling empty arrays
234             valence(i,k) = emo_fdata(i,val_ind);
235             arousal(i,k) = emo_fdata(i,arl_ind);
236             strength(i,k) = emo_fdata(i,strength_ind);
237             displacement(i,k) = emo_fdata(i,disp_ind);
238             time(i,k) = emo_fdata(i,time_ind);
239             velocity(i,k) = emo_fdata(i,vel_ind);
240             acceleration(i,k) = emo_fdata(i,acc_ind);
241             jerk(i,k) = emo_fdata(i,jerk_ind);
242         end
243 
244     end
245 
246 end
247 
248 
249 function [actualBoundary, actualFrame] = timepointCalc(timepoints,
        timeGap, timeStop)
250 
251     % This is the timescale followed by the data
252     timeFrame = 0:timeGap:timeStop;
253 
254     % Find the timepoints of shift
255     % ragaTime  = [45, 125, 157, 192, 258, 366, 442];
256         % This is to calculate the timepoints corresponding to the
            Raga shift
```

```
257         % time points
258
259     actualBoundary = zeros(length(timepoints)+2,1);
260     for i = 1:length(timepoints)
261         [minVal,minInd] = min(abs(timeFrame - timepoints(i)));
262         actualBoundary(i+1) = timeFrame(minInd);
263     end
264     actualBoundary(length(timepoints)+2) = timeFrame(length(timeFrame)
            );
265
266     % Find the frame points of shift
267         % This is to calculate the frames corresponding to the Raga
              shift
268         % time points
269     actualFrame= ones(length(timepoints)+2,1);
270     for i = 1:length(timepoints)
271         [minVal,minInd] = min(abs(timeFrame - timepoints(i)));
272         actualFrame(i+1) = minInd;
273     end
274     actualFrame(length(timepoints)+2) = length(timeFrame);
275
276 end
277
278 function accBinary = binarizeAcc(acceleration, fillWindow, addFrame)
279 % Binarising acceleration time points on basis of zero vs some value
280     % fillWindow will fill any gaps in the acceleration time series
          that
281     % occur for less than 1TR with the previous marker
282
283     % addFrame is used to add a 1TR preceding window to all
          acceleration
284     % events. This was to account for any decision making that would
          go
285     % into the acceleration events
286
287     % Initialise the accBinary variable to store binarised
          acceleration data
288     accBinary = zeros(length(acceleration),width(acceleration));
289     % Loop across participants
290     for j = 1:width(acceleration)
291         % Loop across timepoints
```

```
292         for i = 1:length(acceleration)
293             % Convert all non-zero events into acceleration windows
                    (1)
294             if acceleration(i,j) ~= 0
295                 accBinary(i,j) = 1;
296             else
297             % Convert all zero events into non-acceleration windows
                    (0)
298                 accBinary(i,j) = 0;
299             end
300         end
301     end
302
303     if fillWindow == true
304         % Single occurence windows are converted into preceding event
305         for j = 1:width(acceleration)
306             for i = 2:length(acceleration)-1
307                 if accBinary(i-1,j)==accBinary(i+1,j)
308                     if accBinary(i-1,j)~=accBinary(i,j)
309                         if accBinary(i,j) == 0
310                             accBinary(i,j) = 1;
311                         else
312                             accBinary(i,j) = 0;
313                         end
314                     end
315                 end
316             end
317         end
318     end
319
320     if addFrame == true
321         % Considering that the decision for acceleration precedes
                actual
322         % acceleration, a preceding window is ideal to look into
                underlying
323         % computation.
324
325         % Adding a 1 TR window before every acceleration time point
326         for j = 1:width(acceleration)
327             for i = 2:length(acceleration)
328                 if accBinary(i-1,j)~=accBinary(i,j)
```

```
                    if accBinary(i,j)==1
                        accBinary(i-1,j)=1;
                    end
                end
            end
        end
    end

end

function ind_lat = calcLatency(accBinary, actualFrame, timeBin, divide
    , duration, limit)

    % Initialise empty array
    ind_lat = zeros(8,width(accBinary));

    % List of raga indices
    t_ind = actualFrame;
    % t_ind = [1,39,105,132,161,216,306,369];

    % Initialise loop across participants
    for i = 1:width(accBinary)

        % Loop to index out raga indices and generate raga specific
            tiem frames
        for k = 1:8

            % For the first index there isn't a need to account for
                runovers
            if k==1
                for j = t_ind(k):(t_ind(k+1)-1)
                    % Check if the time point is a transition point or
                        not
                    if accBinary(j,i) == 1
                        if divide == true
                            ind_lat(k,i) = (((j-t_ind(k))+1)*timeBin)/
                                duration(k);
                            if limit == true
                                if ind_lat(k,i) > 32
                                    ind_lat(k,i) = 0;
                                end
```

```
                        end
                    else
                        ind_lat(k,i) = ((j-t_ind(k))+1)*timeBin;
                        if limit == true
                            if ind_lat(k,i) > 32
                                ind_lat(k,i) = 0;
                            end
                        end
                    end
                    break
                end
            end

        % Loop from the second last raga index to the end
        elseif k==8
                % Chcek if the 8th index timepoint and timepoint+1
                    is a transition point
                if accBinary(t_ind(k),i)==1 || accBinary(t_ind(k)
                    +1,i)==1
                % Find the next closest which is a non-transition
                    index
                    % If Kth index is a transition point
                    if accBinary(t_ind(k),i)==1
                        for j = t_ind(k):length(accBinary)
                            % The closest non-transition index
                                becomes
                            % the new starting point.
                            if accBinary(j,i) == 0
                                newInd = j;
                                break
                            end
                        end
                    % If (K+1)th index is a transition point
                    elseif accBinary(t_ind(k)+1,i)==1
                        for j = t_ind(k)+1:length(accBinary)
                            % The closest non-transition index
                                becomes
                            % the new starting point.
                            if accBinary(j,i) == 0
                                newInd = j;
                                break
```

```
                    end
                end
            end

            % Search from this new index for the closest
            % transition point.
            for j = newInd:length(accBinary)
                if accBinary(j,i)==1
                    if divide == true
                        ind_lat(k,i) = (((j-t_ind(k))+1)*
                            timeBin)/duration(k);
                        if limit == true
                            if ind_lat(k,i) > 32
                                ind_lat(k,i) = 0;
                            end
                        end

                    else
                        ind_lat(k,i) = ((j-t_ind(k))+1)*
                            timeBin;
                        if limit == true
                            if ind_lat(k,i) > 32
                                ind_lat(k,i) = 0;
                            end
                        end
                    end
                    break
                end
            end

        % If the Kth and (K+1)th index are not transition
        % points, directly move to calculating latency.
        else
            for j = t_ind(k):length(accBinary)
                if accBinary(j,i) == 1
                    if divide == true
                        ind_lat(k,i) = (((j-t_ind(k))+1)*
                            timeBin)/duration(k);
                        if limit == true
                            if ind_lat(k,i) > 32
                                ind_lat(k,i) = 0;
```

```
                                    end
                                end
                            else
                                ind_lat(k,i) = ((j-t_ind(k))+1)*
                                    timeBin;
                                if limit == true
                                    if ind_lat(k,i) > 32
                                        ind_lat(k,i) = 0;
                                    end
                                end
                            end
                            break
                        end
                    end
                end

            % For other indices
            else
                % Chcek if the 8th index timepoint and timepoint+1 is
                    a transition point
                if accBinary(t_ind(k),i)==1 || accBinary(t_ind(k)+1,i)
                    ==1
                % Find the next closest which is a non-transition
                    index
                    % If Kth index is a transition point
                    if accBinary(t_ind(k),i)==1
                        for j = t_ind(k):t_ind(k+1)
                            % The closest non-transition index becomes
                            % the new starting point.
                            if accBinary(j,i) == 0
                                newInd = j;
                                break
                            end
                        end
                    % If (K+1)th index is a transition point
                    elseif accBinary(t_ind(k)+1,i)==1
                        for j = t_ind(k)+1:t_ind(k+1)
                            % The closest non-transition index becomes
                            % the new starting point.
                            if accBinary(j,i) == 0
                                newInd = j;
```

```
                        break
                    end
                end
            end

            % Search from this new index for the closest
            % transition point.
            for j = newInd:t_ind(k+1)
                if accBinary(j,i)==1
                    if divide == true
                        ind_lat(k,i) = (((j-t_ind(k))+1)*
                            timeBin)/duration(k);
                        if limit == true
                            if ind_lat(k,i) > 32
                                ind_lat(k,i) = 0;
                            end
                        end
                    else
                        ind_lat(k,i) = ((j-t_ind(k))+1)*
                            timeBin;
                        if limit == true
                            if ind_lat(k,i) > 32
                                ind_lat(k,i) = 0;
                            end
                        end
                    end
                    break
                end
            end

        % If the Kth and (K+1)th index are not transition
        % points, directly move to calculating latency.
        else
            for j = t_ind(k):t_ind(k+1)
                if accBinary(j,i) == 1
                    if divide == true
                        ind_lat(k,i) = (((j-t_ind(k))+1)*
                            timeBin)/duration(k);
                        if limit == true
                            if ind_lat(k,i) > 32
                                ind_lat(k,i) = 0;
```

```
                                    end
                                end

                            else
                                ind_lat(k,i) = ((j-t_ind(k))+1)*
                                    timeBin;
                                if limit == true
                                    if ind_lat(k,i) > 32
                                        ind_lat(k,i) = 0;
                                    end
                                end
                            end
                            break
                        end
                    end
                end
            end
        end
    end
end
```

Denoising pre-processed neural data

```
    #%% Import required packages

from os.path import  join, basename
import glob
import pandas as pd
#import numpy as np
import nilearn.image as image
import nibabel as nib
from nilearn.interfaces.fmriprep import load_confounds

#%% Slicing BOLD files

'''
    Removal of the first 13 and last 2 volumes from the raw BOLD data
        (443 volumes)
    to make it a 428 volume time series.
'''

```

```
# Location of EPI files:
data_dir = '/media/austin/Emotion/InterAffectTravel/neuralData/workingBOLD/epi_raw/'

# glob over all possible files with given name pattern in the provided directory:
file_list = sorted(glob.glob(join(data_dir, '*_task-music_space-MNI152NLin2009cAsym_res-2_desc-preproc_bold.nii.gz')))

# Loop through all files in the file list acquired by globbing over given directory:
for f in file_list:
    # Extract subject name:
    sub = basename(f).split('_')[0]
    print(sub)
    # Load the original EPI file
    f_img = image.load_img(f)
    # Index out required volumes and create a new volume
    final_img = image.index_img(f_img,slice(12,440))
    # Save new EPI file
    final_img.to_filename(f)

#%% Slicing the confound files

'''
    Removal of the first 13 and last 2 volumes from the raw confounds data (443 volumes)
    to make it a 428 volume time series.
'''

# Location of confound files:
data_dir = '/media/austin/Emotion/InterAffectTravel/neuralData/workingBOLD/epi_raw/'

# glob over all possible files with given name pattern in the provided directory:
file_list = sorted(glob.glob(join(data_dir,'*_task-music_desc-confounds_timeseries.tsv')))

# Loop through all files in the file list acquired by globbing over given directory:
```

```
for f in file_list:
    # Extract subject name:
    sub = basename(f).split('_')[0]
    print(sub)
    # Load the original confound file
    bold_data = pd.read_csv(f, sep='\t', header = 0)
    # Index out required enteries and create a new matrix
    func_data = bold_data[12:440]
    # Save new confound file
    func_data.to_csv(f'/media/austin/Emotion/InterAffectTravel/
        neuralData/workingBOLD/epi_raw/{sub}_task-music_desc-
        confounds_timeseries.tsv', index=False, sep='\t')

#%% Extracting framewise displacement values and creating a matrix

# Location of confound files:
data_dir = '/media/austin/Emotion/InterAffectTravel/neuralData/
    workingBOLD/epi_raw/'

# glob over all possible files with given name pattern in the provided
     directory:
file_list = sorted(glob.glob(join(data_dir,'*_task-music_desc-
    confounds_timeseries.tsv')))

# Loop through all files in the file list acquired by globbing over
    given directory:
fd = []
for f in file_list:
    # Extract subject name:
    sub = basename(f).split('_')[0]
    print(sub)
    # Load the original confound file
    bold_data = pd.read_csv(f, sep='\t', header = 0)
    # EXtract out the FD columns
    func_data = bold_data.loc[:,'framewise_displacement']
    # Index out required enteries and create a new matrix
    fd.append(func_data)
# Transpose the created matrix into (timepoints,subject):
fd = pd.DataFrame(fd).T
# Rename each column for participant:
fd = fd.set_axis([f'FD_sub-{x:02}' for x in range(1,len(file_list)+1)
```

```
    ], axis=1)
# Save the FD data matrix
fd.to_csv('/media/austin/Emotion/InterAffectTravel/neuralData/
    workingBOLD/fd_allpart.csv', index=False)

#%% Denoising the data, removing the effects of motion variables, CSF,
     and detrending the data

'''
    This script is for loading in the confound files generated by
        fmriprep during preprocessing and regressing them out
    to denoise the BOLD data. It regresses out CSF effects, WM
        activity, motion artifacts captured by Framewise displacement
        and drift regressors from cosine.

'''

# Smoothing kernel
fwhm = 6

# Location of framewise displacement data matrix, to be used in
    confound removal
fd_val = pd.read_csv('/media/austin/Emotion/InterAffectTravel/
    neuralData/workingBOLD/fd_allpart.csv')

# Directory with preprocessed EPI files
base_dir = '/media/austin/Emotion/InterAffectTravel/neuralData/
    workingBOLD/'

# globing over all in specified format in the base_dir
file_list = sorted([x for x in glob.glob(join(base_dir, '*epi_raw/*
    _task-music_space-MNI152NLin2009cAsym_res-2_desc-preproc_bold.nii.
    gz')) if 'denoise' not in x])

# Looping over each file found by glob
for i, f in enumerate(file_list):

    # Extract subject name
    sub = basename(f).split('_')[0]
    print('The following subject data is being denoised:',sub)

```

```
    # Load EPI file
    readBrain = nib.load(f)

    # Smoothen the image according to specified smoothing kernel
    brain_data = image.smooth_img(readBrain, fwhm = fwhm)

    # Load in required confounds, generate a design matrix to regress
        them out
    confounds_simple, sample_mask = load_confounds(
        f,
        strategy=["motion", "wm_csf", "high_pass"],
        motion="derivatives",
        wm_csf="derivatives",
        demean = True # Centering the confounds for each participant,
            Made false
    )

    # Include the framewise displacement data in the confound matrix
    confound_final = pd.concat([confounds_simple,fd_val.iloc[:,i]],
        axis = 1)
    print("The shape of the confound matrix is:", confound_final.shape
        )
    print(confound_final.columns)

    # final_img = clean_img(brain_data, detrend=True, confound =
        confounds_simple, t_r = 1.2, mask_img = brain_mask)
    # Don't load masks ahead of time, nilern GLM instance errors out
        as it can't figure out where to derive information from
    # Detrending removes the influence of proximate voxels on each
        other
    # Regress out each variable in the confounds matrix obtaining the
        final denoised BOLD data:
    final_img = image.clean_img(brain_data, detrend = True, confound =
         confound_final, t_r = 1.2)

    # Save the final processed BOLD file:
    print("Saving denoised data for subject:", sub)
    nib.save(final_img, join(base_dir, 'epi_CosineRemoved', f'{sub}
        _nilearn_denoise_smooth{fwhm}mm_task-music_space-
        MNI152NLin2009cAsym_res-2_desc-preproc_bold.nii.gz'))
```

GLM analysis of the neural data

```
#%% # Creating the acceleration design matrix revamped

def create_event_file(timeseries, tr=1.0):
    """
    Create event file for Nilearn from a binary timeseries.

    Parameters:
    - timeseries: 1D array-like of 0s and 1s
    - tr: Repetition time (in seconds)

    Returns:
    - events: DataFrame with onset, duration, trial_type
    """
    timeseries = np.asarray(timeseries)
    assert set(np.unique(timeseries)).issubset({0, 1}), "Timeseries
       must only contain 0 and 1"

    events = []
    current_val = timeseries[0]
    onset = 0

    for i in range(1, len(timeseries)):
        if timeseries[i] != current_val:
            duration = round((i - onset) * tr,1) - tr
            trial_type = ["NA" if current_val==0 else "A"][0]
            if duration > 0:
                events.append({
                    "onset": round(onset * tr, 1),
                    "duration": duration,
                    "trial_type": trial_type,
                    "modulation" : 1
                })
            onset = i
            current_val = timeseries[i]

    # Add final event
    duration = round((len(timeseries) - onset) * tr, 1) - tr
    trial_type = ["NA" if current_val==0 else "A"][0]
    if duration > 0:
```

```
        events.append({
            "onset": round(onset * tr,1),
            "duration": duration,
            "trial_type": trial_type,
            "modulation" : 1
        })

    return pd.DataFrame(events)

avna = pd.read_csv('/media/austin/Emotion/InterAffectTravel/
    behaviouralData/Data_TRtime/accBinarised_TRt.csv',header=None).T

conditionS = {}
dm_dict = {}

for sub_ind in range(avna.shape[1]):

    conditionS[sub_ind] = create_event_file(avna.iloc[:,sub_ind], tr
        =1.2)

    # Time frame of the whole experiment in seconds, obtained as
        number of EPI volumes x TR time
    frame_times = np.array([round(x*1.2,2) for x in range(428)])
    # Create the first level design matrix.
    ''' Glover HRF model is used for our purposes. This can be changed
         for better fit '''
    dm = make_first_level_design_matrix(frame_times, events =
        conditionS[sub_ind], hrf_model="glover", drift_model=None)
    print(sub_ind)
    # Load generated design matrix to the dictionary with the
        participant ID as a dictionary key
    dm_dict[sub_ind] = dm

# Save design matrix for figure
plot_design_matrix(dm_dict[3])
plt.savefig("/home/austin/Documents/AustinV_Thesis/figures/finalFigure
    /GLM_TransitionCoding_DesignMatrix_Sub3.png", transparent=True)


#%% # First Level Analysis GLM run per participant at voxel

```

```
# Specify the size of the smoothign kernel used for denoising.
# THIS IS ONLY USED TO CORRECTLY READ IN FILES
fwhm=6

# Specify the directory to glob for denoised EPI files
base_dir = '/media/austin/Emotion/InterAffectTravel/neuralData/
    workingBOLD/epi_CosineRemoved/'

# glob over provided directory checking for specified condition
file_list = sorted([x for x in glob.glob(os.path.join(base_dir,f'*
    nilearn_denoise_smooth{fwhm}mm_task-music_space-
    MNI152NLin2009cAsym_res-2_desc-preproc_bold.nii.gz'))])

# Initiliaze a dictionary to store 1st lvl GLM results
sub_dict = {}

# Loop over file list generated by glob
for i,f in enumerate(file_list):

    # Extract subject ID
    sub = os.path.basename(f).split('_')[0]
    print('The following subject data is being processed:',sub)

    # Load in the EPI file
    neuroData = nib.load(f)

    # Call subect specific design matrix, from the dictionary of
        design matrices created above
    dm = dm_dict[i]

    ''' For use only if a whole brain mask template isn't available,
        otherwise stick to using established templates. Check
        templateflow if you can't find something specific'''
    # compute a whole brain mask, voxels within this mask will be
        denoised and extracted.
    #brain_mask = compute_brain_mask(neuroData, mask_type='whole-brain
        ', threshold = 0.08)

    ''' Same as above '''
    # Using an MNI152 brain mask to potentially remove the error
        caused due to different sizes of nifti files and mask
```

```
    #brain_mask = mni_mask(resolution=2) #still throwing error

    # Load in brain mask template
    brain_mask = nib.load('/media/austin/Emotion/InterAffectTravel/
        atlases/tpl-MNI152NLin2009cAsym_res-02_desc-brain_mask.nii.gz')

    # Initiliaze a first level model
    glm_model = FirstLevelModel(t_r = 1.2, hrf_model='glover',
        mask_img=brain_mask, n_jobs=-1, signal_scaling=False)
    # Fit brain data to loaded design matrix
    glm_model = glm_model.fit(neuroData, design_matrices = dm)

    print("Creating a constrast matrix")
    # Intialize an empty contrast array
    contrast_matrix = np.eye(dm.shape[1])

    # Draw contrast values from the design matrix.
    # Column names and values are drawn and autofilled.
    basic_constrats = {
        column: contrast_matrix[i]
        for i, column in enumerate(dm.columns)
        }

    # Setup contrast conditions
    contrasts = {
        "A-NA": basic_constrats["A"] - basic_constrats["NA"],
        "NA-A": basic_constrats["NA"] - basic_constrats["A"],
        "Both": np.vstack((basic_constrats["A"], basic_constrats["NA"
            ]))
        }

    print(f"Computing constrasts for subject: {sub}")

    # Compute the contrasts
    for contrast_id, contrast_val in contrasts.items():
        print(f"\tcontrast id: {contrast_id}")
        #z_map = glm_model.compute_contrast(contrast_val, output_type
            ="z_score") # Use to produce and save only zscored effect
            size
        sub_map = glm_model.compute_contrast(contrast_val, output_type
            ='all')
```

```
        sub_dict[sub+':'+contrast_id] = sub_map # Save to GLM result
            dictionary


#%% # Save the unthresholded data for all participants

''' Unthresholded zmaps will be required for group level GLM '''

# Call subject keys within the dictionary and save the zscored beta
    maps without thresholding
for key,data in sub_dict.items():
    nib.save(sub_dict[key]['z_score'], f'/media/austin/Emotion/
        InterAffectTravel/neuralData/analysis/GLM/fstlvl/{key}
        _unfiltered_zmap.nii.gz') # change z_score to other data types
        to save them


#%% # pThresholding for all participant data

''' To save thresholded zscored effect sizes calculated by the first
    level GLM '''
# Loop across participants
for j in range(file_list.shape[0]):

    # Call subject keys within the dictionary and save the zscored
        beta maps without thresholding
    for key,data in sub_dict.items():

        # Check for participant number and call keys specific to the
            participant
        if f"{j}".zfill(2) in key:

            # Extract the zscored effect size map
            z_map = sub_dict[key]['z_score']

            # beta_map = sub_dict[key]['effect_size'] # Uncomment if
                you require non-zscored effect size values

            # Threshold the stat map(zscored effect size map) with 5%
                FDR and 10 voxel cluster threshold
            pThresh = nilearn.glm.threshold_stats_img(z_map,
```

```
                height_control = 'fdr', alpha = 0.05, cluster_threshold
                =10)[0]
171
172             # Zero out voxels that are insignificant after FDR
                    correction
173             FinImg = math_img("img1*img2", img1 = z_map, img2 =
                    binarize_img(pThresh), copy_header_from = 'img1')
174
175             #plot_stat_map(FinImg, cut_coords=[1,-40,33])
176
177             # Save thresholded stat maps
178             nib.save(FinImg, f'/media/austin/Emotion/InterAffectTravel
                    /neuralData/analysis/GLM/zmap_AVNA_pThresh/{key}
                    _pThresh_zmap.nii.gz')
179
180
181 #%% # Second level analysis, confirmation of group level behaviour in
        the participant pool
182
183 ''' USE UNTHRESHOLED STAT MAPS FOR THIS. DO NOT USE FDR CORRECTED ONES
     AS THEY WILL ERROR OUT'''
184
185 '''
186     The second level GLM uses a design matrix with the subjects as the
          variable under consideration, making it a group level
        comparison.
187     This allows for any group level effects to be extracted and
        observed.
188 '''
189
190 # glob to get location information for all unthresholded stat maps
    saved previously
191 file_list = sorted([x for x in glob.glob(os.path.join('/media/austin/
    Emotion/InterAffectTravel/neuralData/analysis/GLM/fstlvl/sub*A-NA*'
    ))])
192
193 # Intialize empty array
194 second_level_input = []
195
196 # Loop over participant files
197 for i,f in enumerate(file_list):
```

```
198     # Extract subject number
199     sub = os.path.basename(f).split(':')[0]
200     print('The following subject data is being processed:',sub)
201     # Load subject specific uthresholded zmap
202     contData = nib.load(f)
203     # Append to create a cluster of subject specific uthresholded zmap
204     second_level_input.append(contData)
205
206 # Create second level design matrix
207 design_matrix = pd.DataFrame( [1] * len(file_list), columns=["
        intercept"] )
208
209 # Initialize second level GLM
210 glm2model = second_level.SecondLevelModel()
211 # Fit data to second level GLM
212 glm2model = glm2model.fit(second_level_input, design_matrix =
        design_matrix)
213 # Compute group level effects
214 z_map = glm2model.compute_contrast(output_type='all')
215
216 # Threshold using a 1% FDR and 10 voxel cluster threshold.
217 ''' This mask produces the output for a one tailed hypothesis test. It
         can also produce two tailed results where any difference is shown
        by setting two_sided to false '''
218 pMask = nilearn.glm.threshold_stats_img(z_map['z_score'],
        height_control = 'fdr', alpha = 0.05, cluster_threshold=10,
        two_sided=False)[0]
219 # FinImg = math_img("img1*img2", img1 = z_map['z_score'], img2=
        binarize_img(pMask), copy_header_from='img1')
220 #pMask2 = nilearn.glm.threshold_stats_img(pMask, threshold=0,
        height_control = None)[0]
221 #plot_stat_map(z_map['z_score'], cut_coords=[1,-40,33])
222
223 plot_stat_map(pMask, cut_coords=[1,-40,33], title='Regions active in
        emotional tranistions')
224 plt.savefig("/home/austin/Documents/AustinV_Thesis/figures/finalFigure
        /GLM_TransitionCoding_A-NA.png", transparent=True)
225
226 nib.save(pMask, '/media/austin/Emotion/InterAffectTravel/neuralData/
        analysis/GLM/glm2ndLVL_A-NA_pThresh5%_zmap.nii.gz')
```

References

[1] David J. Anderson and Ralph Adolphs. "A Framework for Studying Emotions across Species". In: *Cell* 157.1 (Mar. 2014), pp. 187–200. ISSN: 0092-8674. DOI: 10.1016/j.cell.2014.03.003. URL: http://dx.doi.org/10.1016/j.cell.2014.03.003.

[2] Eeva T. Aronen et al. "Motor activity in depressed children". In: *Journal of Affective Disorders* 133.1–2 (Sept. 2011), pp. 188–196. ISSN: 0165-0327. DOI: 10.1016/j.jad.2011.04.013. URL: http://dx.doi.org/10.1016/j.jad.2011.04.013.

[3] B Avants et al. "Symmetric diffeomorphic image registration with cross-correlation: Evaluating automated labeling of elderly and neurodegenerative brain". In: *Medical Image Analysis* 12.1 (Feb. 2008), pp. 26–41. ISSN: 13618415. DOI: 10.1016/j.media.2007.06.004. URL: https://linkinghub.elsevier.com/retrieve/pii/S1361841507000606.

[4] Yashar Behzadi et al. "A component based noise correction method (CompCor) for BOLD and perfusion based fMRI". In: *NeuroImage* 37.1 (Aug. 1, 2007), pp. 90–101. ISSN: 1053-8119. DOI: 10.1016/j.neuroimage.2007.04.042. URL: https://www.sciencedirect.com/science/article/pii/S1053811907003837.

[5] Niall Bolger et al. "Diary Methods: Capturing Life as it is Lived". In: *Annual Review of Psychology* 54.1 (Feb. 2003), pp. 579–616. ISSN: 1545-2085. DOI: 10.1146/annurev.psych.54.101601.145030. URL: http://dx.doi.org/10.1146/annurev.psych.54.101601.145030.

[6] Brendan P. Bradley et al. "Attentional Biases for Emotional Faces". In: *Cognition & Emotion* 11.1 (Jan. 1997), pp. 25–42. ISSN: 0269-9931, 1464-0600. DOI: 10.1080/026999397380014. URL: http://www.tandfonline.com/doi/abs/10.1080/026999397380014.

[7] Rastko Ciric et al. "TemplateFlow: FAIR-sharing of multi-scale, multi-species brain models". In: *Nature Methods* 19.12 (Dec. 2022). Publisher: Nature Publishing Group, pp. 1568–1571. ISSN: 1548-7105. DOI: 10.1038/s41592-022-01681-2. URL: https://www.nature.com/articles/s41592-022-01681-2.

[8] Nilearn contributors. *nilearn*. DOI: https://doi.org/10.5281/zenodo.8397156. URL: https://github.com/nilearn/nilearn.

[9] Mihaly Csikszentmihalyi et al. "The ecology of adolescent activity and experience". In: *Journal of Youth and Adolescence* 6.3 (Sept. 1977), pp. 281–294. ISSN: 1573-6601. DOI: 10.1007/bf02138940. URL: http://dx.doi.org/10.1007/BF02138940.

[10] Oscar Esteban et al. "fMRIPrep: a robust preprocessing pipeline for functional MRI". In: *Nature Methods* 16.1 (Jan. 2019). Publisher: Nature Publishing Group, pp. 111–116. ISSN: 1548-7105. DOI: 10.1038/s41592-018-0235-4. URL: https://www.nature.com/articles/s41592-018-0235-4.

[11] VS Fonov et al. "Unbiased nonlinear average age-appropriate brain templates from birth to adulthood". In: *NeuroImage*. Organization for Human Brain Mapping 2009 Annual Meeting 47 (July 1, 2009), S102. ISSN: 1053-8119. DOI: 10.1016/S1053-8119(09)70884-5. URL: https://www.sciencedirect.com/science/article/pii/S1053811909708845.

[12] Jessica J. Genet et al. "Flexibility is not always adaptive: Affective flexibility and inflexibility predict rumination use in everyday life". In: *Cognition and Emotion* 27.4 (June 2013), pp. 685–695. ISSN: 1464-0600. DOI: 10.1080/02699931.2012.733351. URL: http://dx.doi.org/10.1080/02699931.2012.733351.

[13] Jutta Globisch et al. "Fear appears fast: Temporal course of startle reflex potentiation in animal fearful subjects". In: *Psychophysiology* 36.1 (1999), pp. 66–75. ISSN: 1469-8986. DOI: 10.1017/S0048577299970634. URL: https://onlinelibrary.wiley.com/doi/abs/10.1017/S0048577299970634.

[14] Gary H. Glover. "Deconvolution of Impulse Response in Event-Related BOLD fMRI1". In: *NeuroImage* 9.4 (Apr. 1999), pp. 416–429. ISSN: 10538119. DOI: 10.1006/nimg.1998.0419. URL: https://linkinghub.elsevier.com/retrieve/pii/S1053811998904190.

[15] Krzysztof Gorgolewski et al. "Nipype: a flexible, lightweight and extensible neuroimaging data processing framework in python". In: *Frontiers in Neuroinformatics* 5 (2011), p. 13. ISSN: 1662-5196. DOI: 10.3389/fninf.2011.00013.

[16] Douglas N. Greve and Bruce Fischl. "Accurate and robust brain image alignment using boundary-based registration". In: *NeuroImage* 48.1 (Oct. 15, 2009), pp. 63–72. ISSN: 1053-8119. DOI: 10.1016/j.neuroimage.2009.06.060. URL: https://www.sciencedirect.com/science/article/pii/S1053811909006752.

[17] James J Gross. “Antecedent-and response-focused emotion regulation: divergent consequences for experience, expression, and physiology.” In: *Journal of personality and social psychology* 74.1 (1998), p. 224.

[18] James J Gross. “The emerging field of emotion regulation: An integrative review”. In: *Review of general psychology* 2.3 (1998), pp. 271–299.

[19] James J. Gross. “The Extended Process Model of Emotion Regulation: Elaborations, Applications, and Future Directions”. In: *Psychological Inquiry* 26.1 (Jan. 2015), pp. 130–137. ISSN: 1532-7965. DOI: 10.1080/1047840x.2015.989751. URL: http://dx.doi.org/10.1080/1047840X.2015.989751.

[20] Christiane A. Hoppmann and Amy Ho. *Ecological Momentary Assessment*. Dec. 2015. DOI: 10.1002/9781118521373.wbeaa102. URL: http://dx.doi.org/10.1002/9781118521373.WBEAA102.

[21] Chloe Hutton et al. “Image Distortion Correction in fMRI: A Quantitative Evaluation”. In: *NeuroImage* 16.1 (May 2002), pp. 217–240. ISSN: 10538119. DOI: 10.1006/nimg.2001.1054. URL: https://linkinghub.elsevier.com/retrieve/pii/S1053811901910547.

[22] Mark Jenkinson, Peter Bannister, et al. “Improved Optimization for the Robust and Accurate Linear Registration and Motion Correction of Brain Images”. In: *NeuroImage* 17.2 (Oct. 1, 2002), pp. 825–841. ISSN: 1053-8119. DOI: 10.1006/nimg.2002.1132. URL: https://www.sciencedirect.com/science/article/pii/S1053811902911328.

[23] Mark Jenkinson and Stephen Smith. “A global optimisation method for robust affine registration of brain images”. In: *Medical Image Analysis* 5.2 (June 1, 2001), pp. 143–156. ISSN: 1361-8415. DOI: 10.1016/S1361-8415(01)00036-6. URL: https://www.sciencedirect.com/science/article/pii/S1361841501000366.

[24] Peter Kuppens, Jeroen Stouten, et al. “Individual differences in emotion components and dynamics: Introduction to the Special Issue”. In: *Cognition and Emotion* 23.7 (Nov. 2009), pp. 1249–1258. ISSN: 0269-9931, 1464-0600. DOI: 10.1080/02699930902985605. URL: https://www.tandfonline.com/doi/full/10.1080/02699930902985605.

[25] Peter Kuppens and Philippe Verduyn. “Looking at Emotion Regulation Through the Window of Emotion Dynamics”. In: *Psychological Inquiry* 26.1 (Jan. 2, 2015), pp. 72–79. ISSN: 1047-840X, 1532-7965. DOI: 10.1080/1047840X.2015.960505. URL: http://www.tandfonline.com/doi/abs/10.1080/1047840X.2015.960505.

[26] Ashley M. Malooly et al. “Individual differences in reappraisal effectiveness: The role of affective flexibility.” In: *Emotion* 13.2 (2013), pp. 302–313. ISSN: 1528-3542. DOI: 10.1037/a0029980. URL: http://dx.doi.org/10.1037/a0029980.

[27] Avantika Mathur et al. “Emotional responses to Hindustani raga music: the role of musical structure”. In: *Frontiers in Psychology* 6 (Apr. 30, 2015). Publisher: Frontiers. ISSN: 1664-1078. DOI: 10.3389/fpsyg.2015.00513. URL: https://www.frontiersin.org/journals/psychology/articles/10.3389/fpsyg.2015.00513/full.

[28] Rémi Patriat et al. “An improved model of motion-related signal changes in fMRI”. In: *NeuroImage* 144 (Jan. 1, 2017), pp. 74–82. ISSN: 1053-8119. DOI: 10.1016/j.neuroimage.2016.08.051. URL: https://www.sciencedirect.com/science/article/pii/S1053811916304360.

[29] Luiz Pessoa. “A Network Model of the Emotional Brain”. In: *Trends in Cognitive Sciences* 21.5 (May 2017), pp. 357–371. ISSN: 1364-6613. DOI: 10.1016/j.tics.2017.03.002. URL: http://dx.doi.org/10.1016/j.tics.2017.03.002.

[30] Jonathan D. Power et al. “Methods to detect, characterize, and remove motion artifact in resting state fMRI”. In: *NeuroImage* 84 (Jan. 1, 2014), pp. 320–341. ISSN: 1095-9572. DOI: 10.1016/j.neuroimage.2013.08.048.

[31] Nadja Razavi et al. “Measuring motor activity in major depression: The association between the Hamilton Depression Rating Scale and actigraphy”. In: *Psychiatry Research* 190.2–3 (Dec. 2011), pp. 212–216. ISSN: 0165-1781. DOI: 10.1016/j.psychres.2011.05.028. URL: http://dx.doi.org/10.1016/j.psychres.2011.05.028.

[32] James A. Russell. “A circumplex model of affect.” In: *Journal of Personality and Social Psychology* 39.6 (Dec. 1980), pp. 1161–1178. ISSN: 1939-1315, 0022-3514. DOI: 10.1037/h0077714. URL: https://doi.apa.org/doi/10.1037/h0077714.

[33] David Sander et al. “A systems approach to appraisal mechanisms in emotion”. In: *Neural Networks* 18.4 (May 2005), pp. 317–352. ISSN: 08936080. DOI: 10.1016/j.neunet.2005.03.001. URL: https://linkinghub.elsevier.com/retrieve/pii/S0893608005000341.

[34] Theodore D. Satterthwaite et al. “An improved framework for confound regression and filtering for control of motion artifact in the preprocessing of resting-state functional connectivity data”. In: *NeuroImage* 64 (Jan. 1, 2013), pp. 240–256. ISSN: 1053-8119. DOI: 10.1016/j.neuroimage.2012.08.052. URL: https://www.sciencedirect.com/science/article/pii/S1053811912008609.

[35] C. Spearman. “The Proof and Measurement of Association between Two Things”. In: *The American Journal of Psychology* 15.1 (Jan. 1904), p. 72. ISSN: 00029556. DOI: 10.2307/1412159. URL: https://www.jstor.org/stable/1412159?origin=crossref.

[36] Shashwati Tripathi et al. “Network features reflect the effect of neuropsychiatric traits in emotion regulation”. In preparation. 2025.

[37] Nicholas J. Tustison et al. “N4ITK: Improved N3 Bias Correction”. In: *IEEE Transactions on Medical Imaging* 29.6 (June 2010), pp. 1310–1320. ISSN: 1558-254X. DOI: 10.1109/TMI.2010.2046908. URL: https://ieeexplore.ieee.org/document/5445030.

[38] Lucina Q. Uddin. “Cognitive and behavioural flexibility: neural mechanisms and clinical considerations”. In: *Nature Reviews Neuroscience* 22.3 (Feb. 2021), pp. 167–179. ISSN: 1471-0048. DOI: 10.1038/s41583-021-00428-w. URL: http://dx.doi.org/10.1038/s41583-021-00428-w.

[39] Pauli Virtanen et al. “SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python”. In: *Nature Methods* 17 (2020), pp. 261–272. DOI: 10.1038/s41592-019-0686-2.

[40] Patrik Vuilleumier, Jorge L. Armony, et al. “Effects of Attention and Emotion on Face Processing in the Human Brain: An Event-Related fMRI Study”. In: *Neuron* 30.3 (May 1, 2001), pp. 829–841. ISSN: 0896-6273. DOI: 10.1016/S0896-6273(01)00328-2. URL: https://www.sciencedirect.com/science/article/pii/S0896627301003282.

[41] Patrik Vuilleumier, Mark P. Richardson, et al. “Distant influences of amygdala lesion on visual cortical activation during emotional face processing”. In: *Nature Neuroscience* 7.11 (Nov. 2004). Publisher: Nature Publishing Group, pp. 1271–1278. ISSN: 1546-1726. DOI: 10.1038/nn1341. URL: https://www.nature.com/articles/nn1341.

[42] Patrik Vuilleumier and Sophie Schwartz. “Emotional facial expressions capture attention”. In: *Neurology* 56.2 (Jan. 23, 2001). Publisher: Wolters Kluwer, pp. 153–158. DOI: 10.1212/WNL.56.2.153. URL: https://www.neurology.org/doi/full/10.1212/wnl.56.2.153.

[43] Y. Zhang et al. “Segmentation of brain MR images through a hidden Markov random field model and the expectation-maximization algorithm”. In: *IEEE Transactions on Medical Imaging* 20.1 (Jan. 2001), pp. 45–57. ISSN: 1558-254X. DOI: 10.1109/42.906424. URL: https://ieeexplore.ieee.org/document/906424.